



TAMPEREEN TEKNILLINEN YLIOPISTO

JOHANNES SULKAKOSKI
RT-KORTISTON BETONIPINNAN TASAISUUDEN MITTAUKSEN
AUTOMATISOINTI
Diplomityö

Tarkastaja: Professori Jukka Vanhala
Tarkastaja ja aihe hyväksytty
Automaatio-, kone- ja
materiaalitekniikan
tiedekuntaneuvoston kokouksessa
3. Lokakuuta 2012

TIIVISTELMÄ

TAMPEREEN TEKNILLINEN YLIOPISTO

Automaatiotekniikan koulutusohjelma

SULKAKOSKI, JOHANNES: RT-Kortiston betonipinnan tasaisuuden mittauksen automatisointi

Diplomityö, 42 sivua, 12 liitesivua

Helmikuu 2013

Pääaine: Sulautetut järjestelmät

Tarkastaja: Professori Jukka Vanhala

Avainsanat: RT-Kortisto, betonipinta, tasaisuuden mittaus, sähköinen

RT-Kortistossa on määritelty betonipinnan tasaisuuden mittaamisen menetelmä, jossa mittaustapahtuma suoritetaan mekaanisesti yksinkertaisella asteikollisella kappaleella. Tulokset kirjataan paperille ja ne siirretään jälkeensä tietokoneelle tarpeen mukaan. Mittausten määrää rajoittaa mittaustapahtuman vaatima työmäärä. Käytännössä pinnan tasaisuuden mittauksen suorittamisen ja sen tulosten kirjaamisen suorittavat eri henkilöt, sillä mittalaudan laskeminen käsistä esimerkiksi seiniä tai kattoja mitatessa on hankalaa. Tämä työ pyrkii automatisoimaan mittaustapahtumasta tulosten kirjaamisen, nopeuttamaan mittauksia niin, että rakennuksen pintojen tutkimiseen olisi matalampi kynnyks ja selvittämään voiko mittaukset tehdä resistiivisellä anturilla.

Valittiin anturityyppi, joka teoriassa soveltui käyttöympäristöön, ja todettiin sen toimivuus käytännön kokein. Tämän jälkeen valmistettiin laite, joka kykeni käyttämään valittua anturityyppiä ja kirjaamaan elektronisesti mittauksista saadut tulokset. Tämän mittalaitteprototyypin oli myös oltava rakenteeltaan sellainen, että sitä saattoi yksi mies kuljettaa sekä käyttää samalla tavoin kuin vanhan menetelmän laitetta.

Kehitetyn mittalaitteprototyypin ja sen alustavien käyttökokeiden tulosten johdosta todetaan, että sekä mittalaitteen konsepti että valittu anturi soveltuvat tehtävään. Mittaustarkkuus on yksittäiselle mittauspisteelle $\pm 0,28\text{mm}$. Resistiivisen anturin käytössä todettiin joitakin varauksia, mutta niiden ei todettu hankaloittavan mittaustapahtumaa niin, etteikö tehdyn mittalaitteprototyypin kaltaista laitetta voitaisi kaupallistaa.

Vähemmän ympäristön lämpötilamuutoksille herkkien anturivaihtoehtojen tutkimista todettiin kuitenkin tarvittavan. Myöskin todettiin tutkitun mittalaitteen todennäköisesti hyötyvän kiihtyvyysanturista lisääntyneinä potentiaalisina käyttökohteina. Lisäksi kerätyn tiedon prosessointia ja esitystä jo mittaustapahtuman aikana on tutkittava, sekä laitteen omalla näytöllä että muulle laitteelle, kuten älypuhelimelle, langattomasti siirrettynä.

ABSTRACT

TAMPERE UNIVERSITY OF TECHNOLOGY

Master's Degree Programme in Automation Technology

SULKAKOSKI, JOHANNES: Modernising the Construction Standards Index

-mandated manual concrete surface evenness measurement

Master of Science Thesis, 42 pages, 12 Appendix pages

February 2013

Major: Embedded Systems

Examiner: Professor Jukka Vanhala

Keywords: construction standards, concrete, evenness measurement

The Index of Construction Standards used in the Finnish construction industry dictates that the evenness of concrete surfaces, for the purposes of roughly defining the class of surface quality, shall be measured with a straight aluminium board and a scaled triangular piece. The results thus obtained are marked on paper and in the current process in use later transferred manually to a computer for further processing and reporting. This manual process is prone to human error and inefficient. In the worst case, the measuring of a ceiling, two persons are usually employed. In this thesis, an attempt was made to develop a device for automating the recording of measurement results. Additionally the use of simple and relatively cheap resistive sensors was attempted.

A proof-of-concept prototype device was developed. This device used the resistive sensors for measurement and a memory card to store results. It mimicked the old manual system in construction and usage as much as possible. The finished device was field-tested. Additionally, the sensors were shown to possess sufficient manufacturer-defined accuracy and were tested for repeatability of results in the projected use.

Although field testing was cut short and revealed a software error, the device as a whole was proven viable. It still needs usability development, but was shown to be a viable replacement to the old manual method. Additionally, with a software calibration method in place, the worst-case accuracy of the resistive sensors used in the prototype device was shown to be $\pm 0,28\text{mm}$ for each measurement point. This was deemed accurate enough for the proposed use cases.

Further study needs were recognized for more accurate and less sensitive to moisture sensors. Adding additional sensors, such as a three-axis accelerometer, were suggested and are likely to be studied by the company that financed this study. Lastly, methods for processing the gathered data on the scene and displaying via better user interface or wirelessly transmitting to a smartphone for analysis were recommended as future study subjects also.

ALKUSANAT

Kiitokset isälle, äidille, lemmikille, JS Piirto Oy:lle, tarkastavalle professorille (erityisesti, kärsivällisyydestä ja pitkästä muistista), sekä erinäisille laitoshenkilökunnan edustajille kärsivällisyydestä.

-JS

SISÄLLYS

1	Johdanto.....	1
2	Prototyypilaitteen kehitys.....	3
2.1	Käyttöliittymä.....	3
2.2	Tulosten kirjaus.....	4
2.3	Anturoinnin levyjen kehitys.....	5
2.4	Laitteen runko ja kotelointi.....	7
3	Prototyypilaitte.....	9
3.1	Yleisrakenne.....	9
3.2	Anturien sijoittelu.....	10
3.3	Mittalaitteen anturien fyysisen kytkentä.....	10
3.4	Anturointipiirilevy	11
3.5	Laitteen käyttöjännite	13
3.6	Mekaaninen suojaus	14
3.7	Käyttöliittymäkomponentit.....	15
4	Prototyypilaitteen ohjelmisto.....	18
4.1	Käyttöliittymä.....	19
4.2	Asetusten ja kalibraatioarvojen talletus.....	22
4.3	Muistikortille kirjoitus.....	22
4.4	Anturoinnin ohjelma.....	23
5	Resistiivinen etäisyysanturi.....	26
5.1	Anturin tyypin valintaperusteet.....	26
5.2	Anturin malli ja kuvaus.....	26
5.3	Mittaustulosten tarkkuus.....	27
5.4	Mittausten toistettavuus.....	29
6	Prototyypilaitteen kenttätestit ja tulokset.....	32
6.1	Ensimmäinen kenttäkoe – Teollisuushalli Korpilahdella.....	32
6.1.1	Mittaustulokset.....	33
6.1.2	Mittaustulosten analyysi.....	33
6.1.3	Käyttötestin palaute.....	34
6.1.4	Tulosten aiheuttamat toimenpiteet.....	34
6.2	Toinen käyttötesti – Toimistokiinteistön kellaritiloja Jyväskylässä.....	34
6.2.1	Mittaustulokset.....	36
6.2.2	Mittaustulosten analyysi.....	36
6.2.3	Käytettävyysanalyysi.....	37
6.2.4	Tulosten aiheuttamat toimenpiteet.....	37
7	Tulosten analysointi.....	38
7.1	JS-Piirto Oy:n huomioita.....	38
7.2	Päätelmät.....	38

7.3	Jatkokehitys.....	39
	Lähteet.....	41
	Liite A.....	43
	Liite B.....	45
	Liite C.....	46
	Liite D.....	47
	Liite E.....	49
	Liite F.....	52
	Liite G.....	53

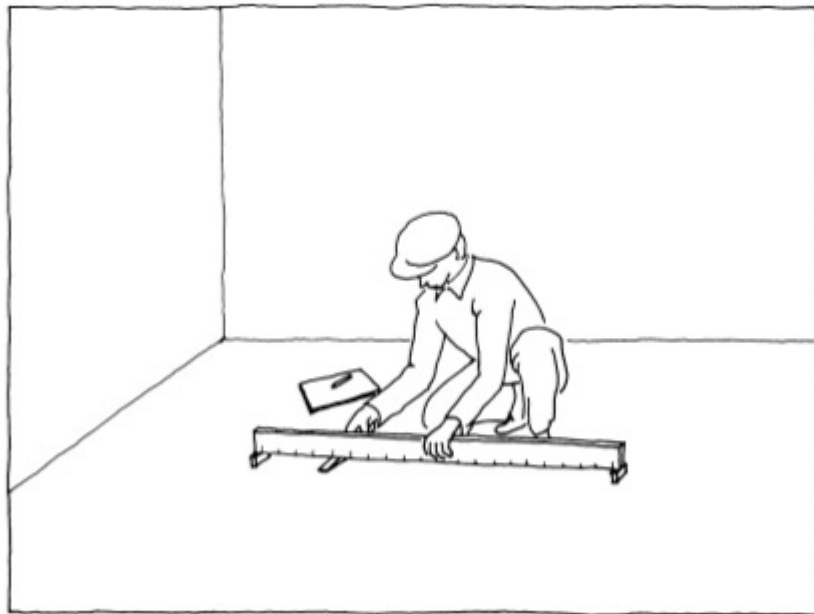
TERMIT JA NIIDEN MÄÄRITELMÄT

RT-kortisto	RakennusTekninen kortisto. Eräänlainen rakentamisen standardinmenetelmien kirjasto, jota ylläpitää Rakennustietosäätiö. Katso [1].
I2C-väylä	Philips-yhtiön kehittämä mikropiirien välinen sarjamuotoinen tiedonsiitoväylä. Tässä työssä sama kuin TWI.
TWI	Two Wire Interface. Lähes sama kuin I2C, mutta kiertää Philipsin nimisuoja.
CSV	Comma Separated Values. Merkintätapa tietokoneen tiedostossa; tiedueet erotetaan toisillaan rivinvaihdolla ja arvot pilkulla. Tyypillisesti CSV-tiedostossa olevat desimaaliluvut käyttävät desimaalipistettä.
AD-muunnos	Analogia-Digitaalimuunnos. Näytteen ottaminen analogisesta lähteestä ja esittäminen diskreettinä numeroarvona.
ADC	Analog-Digital Conversion/Converter. Conversion sama kuin AD-muunnos. Converter on muunnoksen tekevä laitteisto.
AVR	Atmel Co.:n kehittämä 8-bittinen mikrokontrolleriarkkitehtuuri ja -sarja. Kauppanimi.
SPI	Serial Peripheral Interface. Piirilevyn mikropiirien väliseen kommunikointiin tarkoitettu digitaalinen väylä.
MISO	Master In Slave Out. Tiedonsiirtoa ohjaavan piirin sisäänvalo. Toisen osapuolen ulostulo.
MOSI	Master Out Slave In. Tiedonsiirtoa ohjaavan piirin ulostulo. Toisen osapuolen sisäänvalo.
HD 44700	Hitachi Co.:n kehittämä standardi LCD-näytön ja mikroprosessorin kommunikointiin.

1 JOHDANTO

Vuonna 2010 JS-Piirto Oy lähestyi tuttua TTY:n opiskelijaa ja kysyi mahdollisuutta valmistaa laite, joka automatisoisi RT-kortistossa mainitun betonipinnan tasaisuuden mittauksen. Kyseiseen projektiin ei tuolloin ollut aikaa eikä varattuja resursseja. Myöhemmin, vuonna 2012, kyseinen opiskelija oli etsimässä diplomityöpaikkaa. Saman yrityksen edustaja ehdotti saman tutkimustyön ottamista diplomityöksi. Ehdotus hyväksyttiin.

RT-kortiston väline betonipinnan tasaisuuden mittaukseen koostuu kahdesta osasta. Nämä osat ovat suora lauta (ns. mittalauta), jonka päissä on 20mm:n korokepalat, sekä mittakiila, joka on puukiila, jonka pitkällä sivulla on yhden millimetrin koronmuutoksen välein merkintä. [1] Tämä mittaustapa vaatii suorittajansa molemmat kädet, mikä johtaa joko mittaustapahtuman hidastumiseen tai avustajan käyttöön tulosten kirjaamiseksi, mikä puolestaan lisää palkkakustannuksia. Erityistä tarkkuutta vaativiin mittauksiin on lisäksi liikutettavalla mittakellolla varustettu lauta. Tulosten kirjaaminen tapahtuu täysin käsin, mikä voi usein johtaa inhimillisiin virheisiin.



Kuva 1: Perinteinen mittaustapa (Lähde: [1])

Koska suurin hidaste ja virheiden lähde on mittaustulosten kirjaaminen, pyrittiin diplomityössä tekemään mittaustapahtumasta paperiton. Toisin sanoen, mittalaudan korvaava laite suorittaisi oikein asemoituna automaattisesti mittauksen ja kirjaisi tulokset elektronisesti. Tällä laitteella ei pyritä korvaamaan mittakellolla suoritettavaa tarkkuusmittausta vaan nimenomaan kiilapalalla tehtävä karkea mittaus, joka on yleisempi. Tulosten purkuun olisi sallittavaa käyttää erikoisohjelmaa, sillä useat

nykyiset talonrakennuksessa käytettävät mittalaitteet vaativat tämän myös. Ehdotetun mittaukseen käytettävän sensorin ympärille oli myös tehtävä toimiva prototyyppi. Toimivuus todettaisiin koemittauksin.

2 PROTOTYYPPILAITTEEN KEHITYS

Tämä kappale kertoo lyhyesti prototyyppilaitteen kehitysvaiheista. Anturin tyyppi oli valittu jo ennen kehityksen virallista aloitusta, sillä aihetta oli pohdittu idean ensimmäisestä ilmaisusta asti. Anturin valintaperusteista enemmän kappaleessa 5. Myös laitteen tuleva rakenne oli vanhan menetelmän ja tilaavan yrityksen toiveiden pohjalta ilmeinen. Lopullinen rakenne on selvitetty kappaleessa 3.

2.1 Käyttöliittymä

Prototyyppilaitteen suurin epävarmuustekijä oli käyttöliittymä. Laitteen käyttöliittymän piti olla riittävän yksinkertainen, jotta sitä kykeni käyttämään ilman huomattavaa koulutuspanostusta. Toisaalta tuloksia oli voitava kirjata useasta mittaustapahtumasta ilman että oli vaaraa niiden sekoittumisesta. Kaiken tämän lisäksi tulokset oli myös voitava purkaa tietokoneelle ja muutettava ihmisen luettavaan muotoon.

Näytöksi valikoitui tässä vaiheessa alfanumeerinen kaksi- tai useampirivinen LCD näyttö, koska laitteessa tuli potentiaalisesti olemaan huomattava määrä säädettäviä ominaisuuksia, joiden tarkkaa lukumäärää ei voitu vielä alussa tietää; on täysin mahdollista, että useampien kenttäkokeiden jälkeen laitteesta löytyy aina parannettavaa. Samoin hallintalaitteiston pohjaksi valikoitui 3x3-matriisinäppäimistö, jonka arvioitiin olevan riittävän suuri kaikkiin tarpeisiin.

Käyttöliittymäyksikön pohjaksi valikoitui Arduino-yhteensopiva mikrokontrollerialusta. Tälle oli muutamia eri syitä. Arduinon käytöstä laiteprototyypin pohjana oli aikaisempaa kokemusta. Sille oli lisäksi olemassa melko kattavat oheislaitteikirjastot kehitystä aloitettaessa. Arduinossa oleva AVR-mikrokontrolleri on lisäksi virrankulutukseltaan pieni; itse prosessori kuluttaa vain joitakin milliampeereja. [2] [3] Valintaan vaikutti osaltaan myös se, että Arduino-alustaa oli saatavilla jo valmiiksi joitakin kappaleita, jolloin myös käyttöliittymän kehitys voitiin aloittaa heti.

Alfanumeeriset LCD-näytöt, jotka seuraavat Hitachin ohjainyksikköstandardia, vaativat 6 digitaalista lähtöä. [4] Matriisinäppäimistö vaati 6 pinniä ja kommunikaatio muun laitteiston kanssa vaatisi todennäköisesti vähintään 2 pinniä. Lisäksi sarjaliikenneväylä debug-käyttöön ja mahdollisesti tiedon lataamiseen tietokoneelle vaatii 2 pinniä. Laskutoimituksen tuloksena oli todettava, että Arduinossa ei ollut tarpeeksi digitaalisia I/O-pinnejä. Saatavilla oli kuitenkin myös Arduino Mega -klooni, jonka oli valmistanut SeeedStudio. Siinä oli huomattavasti enemmän I/O-pinnejä, minkä lisäksi siinä oli suurempi ohjelmamuisti. Näistä syistä prototyyppilaitteen käyttöliittymän alustaksi otettiin Seeeduino Mega. [5]

Käyttöliittymän ohjelmointiin kului aikaa noin kahden miestyökuukauden verran työtä. Käyttöliittymästä päätettiin ensin tehdä tilakonetyyppinen, jossa ohjelman

suoritus kävisi säännöllisesti näyttöä päivittävässä funktiossa. Tämä osoittautui kuitenkin huonoksi valinnaksi, sillä käyttöliittymän eri valikoiden toiminta on käytännön tarpeiden vuoksi epähomogeenista. Lisäksi näppäimistön tarkastelu hidastuu ja käyttöliittymän vasteaika sen mukana kasvaa. Tästä syystä käyttöliittymäohjelmiston teko päätettiin aloittaa alusta niin, että jokainen käyttöliittymän valikko on oma funktionsa, jossa ohjelman suoritus pysähtyy odottamaan käyttäjän toimintaa. Tällä rakenteella saatiin melko nopeasti lupaavia tuloksia. Käyttöliittymän yhteisille funktioille, kuten ohjelmamuistista tulostamiselle, kirjoitettiin erillinen kirjasto.

Kun käyttöliittymän perusrakenne oli valmis, lisätoimintoja tehtiin tarpeen mukaan. Tämä on yleisesti ohjelmistotuotannossa huono tapa toimia. Ohjelmiston tekoa aloitettaessa kaikki lopputuotteen vaatimukset eivät kuitenkaan olleet tiedossa, joten parhaaksi vaihtoehdoksi katsottiin mahdollisimman hyvän laajennettavuuden säilyttäminen ohjelmiston kehityksen edetessä.

2.2 Tulosten kirjaus

Käyttöliittymän jälkeen ja osin myös samaan aikaan, sillä kyse on samasta ohjelmistosta, huomio käännettiin tulosten kirjaamiseen. Tulosten kirjaamiseen ja niiden saamiseen ulos mittalaitteesta harkittiin kahta eri vaihtoehtoa.

Ensimmäinen vaihtoehto oli tulosten tallettaminen laitteen omaan EEPROM-muistiin, josta ne olisi otettava ulos USB-kaapelin kautta. Tulosten purkamiseen laitteesta olisi tällöin kirjoitettava oma ohjelmistonsa, sillä käyttäjän ei voida edellyttää osaavan käyttää USB-simuloituja sarjaliikenneportteja. Arduino-alustan Atmega1280-mikrokontrollerilla on 4KB sisäistä EEPROM-muistia. [2]

Toinen vaihtoehto oli käyttää ulkoista muistia. Koska Arduino-alustalle on olemassa SD-muistikortille kirjasto, oli tämän korttityypin käyttö helpoin vaihtoehto ulkoiseksi muistiksi. Käytön helppoutta lisäsi se seikka, että mainittu kirjasto tuki suoraan FAT-tiedostojärjestelmää, mikä mahdollisti tiedon lukemisen yleisimmillä PC-käyttöjärjestelmillä kortilta ilman erillistä ohjelmaa.

Mikäli oletetaan mittaustarkkuudeksi enintään 16 bittiä, mittapistettä oletetaan olevan 9 ja tietoa ei pakata tai muuteta ihmisen luettavaan muotoon ennen tallentamista, kuluu yhteen mittaukseen 18 tavua. Mittauksia voitaisiin siis tallettaa Atmega1280-mikrokontrollerin EEPROM-muistiin enintään 222. Tähän lukuun ei sisälly metatieto, kuten mittauskohde tai -rivi, minkä lisäksi osan tilasta oletettiin kuluvan erilaisiin kalibrointi- ja asetustietoihin. Mittauskertojen määrä saattaisi riittää minkä tahansa järkevän kokoisen kohteen mittaamiseen, mutta mikäli samana päivänä tehdään useampia mittauksia, täytyisi data purkaa tietokoneelle kohteiden välissä, mikä voi olla rakennustyömaaolosuhteissa haastavaa. Lisäksi joko tietokone täytyisi kuljettaa mittalaitteen luo tai mittalaite tietokoneen luo, mikä on hyvin epäkäytännöllistä kun

huomioidaan mittalaitteen koko. Paras ratkaisu oli siis käyttää SD-muistikorttia, siten että se on irrotettavissa laitteesta lukemista varten.

SD-kortista on kolme eri kokoa ja kahta eri elektronisesti eroavaa päätyyppiä. Koot ovat täyskoko, minikoko ja mikrokoko joka tavallisemmin tunnetaan nimellä microSD. Elektronisesti eroavat SD ja SDHC, joista jälkimmäinen on suuri kapasiteettisempi ja nopeampi, mutta tarvitsee suuremman väylänopeuden vuoksi aktiivisen tasomuunnoksen (ks. 3.5 – Laitteen käyttöjännite). Tästä syystä päätettiin käyttää normaalinopeuksista SD-korttia. [6]

Mittausdatan tallennusmuodoksi haluttiin ensisijaisesti jokin ihmisen ymmärrettävä muoto. Tämä vaatimus asetettiin jotta tieto olisi mahdollisimman helposti siirrettävissä muihin järjestelmiin sekä myöskin ihmisen tarkasteltavissa mahdollisten virheiden löytämiseksi. Yksinkertaisin tällainen muoto on CSV. Sen lisäksi, että CSV-tiedostot ovat ihmisen ymmärrettävissä, ne on myös mahdollista kirjoittaa datan kertyessä ilman että koko sisältöä tunnetaan etukäteen. CSV-muotoa ymmärtävät myös yleisimmät taulukkolaskentaohjelmat, mikä oli hyödyllistä, sillä näin tiedon käsittelylle ei tarvinnut tehdä erillistä ohjelmistoa. Taulukkolaskentaohjelman käyttömahdollisuus tiedon jatkokäsittelyyn oletettavasti myös vähentää koulutustarvetta laitteen käyttöönotossa.

2.3 Anturoinnin levyjen kehitys

Kun anturien tyyppi oli valittu, alettiin suunnitella tapaa mitata sitä ja saada mittaustulos tallettavalle yksikölle asti. Anturin valintaperusteet ovat kappaleessa 4. Koska mittapisteitä oli paljon ja niiden välimatka oli ulkokäyttöön tulevaan laitteeseen melko pitkä, päätettiin mittaustulokset siirtää digitaalisesti yhteistä väylää pitkin. Analoginen signaali olisi vaatinut vahvistuksen ja häiriöitä vaimentavan kytkennän. Tämän lisäksi mittapisteiden yksittäin kytkeminen olisi kasvattanut kaapeloinnin monimutkaisuutta tarpeettomasti.

Siirtoväyläksi harkittiin kenttäväylää, mutta tästä luovuttiin, sillä halvatkin ratkaisut olisivat tuoneet ylimääräistä laitteistoa ja siten ylimääräisiä kustannuksia. Anturin lukemiseen tehtävän piirilevyn kustannukset ovat sikäli kriittisiä, että näitä levyjä tulee mittalaitteessa olemaan yhtä monta kuin mittaustureitakin. Niiden kustannus siis moninkertaistuu. Tiedonsiirron olisi siis oltava laitteistoltaan edullista. Parhaana vaihtoehtona nähtiin I2C-väylä [7], jota on muuallakin käytetty laitteiden yhdistämiseen jopa useiden metrien kaapelein, kuitenkin harvemmin useampaa kuin kahta laitetta [8]. I2C on kyllä tehty nimenomaan useamman piirin jaetuksi väyläksi, mutta vain piirilevytasolla, ei niiden välillä.

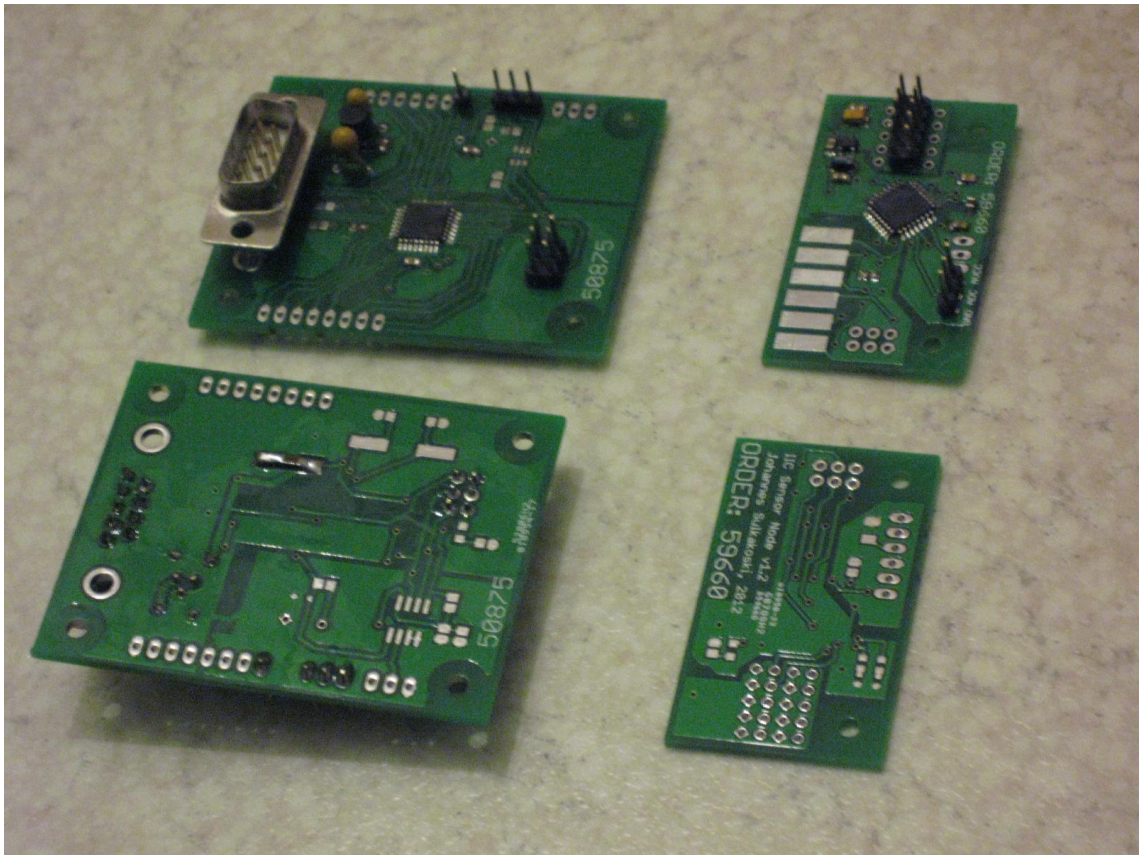
Myös anturia mittaavan AD-muuntimen valinta oli tässä kohtaa vielä avoin. Parhaana vaihtoehtona nähtiin I2C-väylään liitettävää AD-muunnin, sillä näin saataisiin kaikki AD-muuntimet liitettävä samaan väylään suoraan. Ongelmaksi muodostui, että

samanlaisia ADC-piirejä ei saatu eri I2C-osoittein, ainakaan siinä hintaluokassa kuin niitä olisi järkevää tähän tarkoitukseen käyttää. Lisäksi ei ollut varmuutta koko I2C-väylän toimivuudesta. Lopulta päätettiin testata eri vaihtoehtoja sekä kommunikaation että AD-muuntimien suhteen tekemällä prototyyppilevyt, joissa oli kommunikaatiota varten Atmega48V-mikrokontrolleri. Levyllä oli paikat kahdelle sopivalta vaikuttaneelle 12-bittiselle ADC:lle, jotka yhdistettiin mikrokontrolleriin SPI-väylällä. [9]

ATmega48V valittiin, koska siinä on sisäinen I2C-lohko, jolla voitiin kokeilla tämän väylän käyttöä mittapisteiden ja tulokset tallentavan yksikön välillä. Lohkon I2C-osoite voidaan asettaa ohjelmallisesti, joten osoitteiden saanti ei olisi ongelma. Lisäksi, mikäli tämä väylä olisi osoittautunut huonoksi vaihtoehdoksi, mikrokontrollerilla olisi ollut mahdollista kehittää jokin muu kommunikaatioprotokolla, sillä kaikki sen I/O-pinnit olivat yleiskäyttöisiä. Siinä oli lisäksi tarpeeksi laskentatehoa, jotta tuloksia olisi voitu käsitellä ennen lähettämistä, mikäli tarpeen.

Tiedonsiirtoväylä toteutettiin fyysisesti lattakaapelilla, johon anturointilevyt liitettiin D9-liittimin. Käytetty lattakaapeli oli 9-johtiminen, joten kommunikaatiotähtäyksen johtimet oli mahdollista limittää maajohtimien kanssa, mikä myös tehtiin. Tämä suojaa kommunikaatiotähtäystä hieman ulkoisilta häiriöiltä ja enemmän väylän johtimien ristiinkuulumiselta. D9-liitin osoittautui kuitenkin huonoksi valinnaksi, sillä se oli yhdistettynä liitinparina lähes 20mm korkea piirilevyn pinnasta mitattuna.

Tätä ensimmäistä piirilevymallia ei koskaan käytetty. Sen analogisen osan erotus digitaalisen puolen häiriöiltä huomattiin koesarjan valmistuttamisen jälkeen puutteelliseksi. Myös sen jännitelähteen todettiin tuottavan liikaa häiriötä. Lisäksi mittaustarkkuuden vaatimusten tarkennuttua kävi ilmi, että mikrokontrollerin oman ADC-lohkon resoluutio riittää, kun käytetään kalibrointimetodia, joka on selitetty kappaleessa 4. Ensimmäisen ja toisen sarjan piirilevyt nähtävissä kuvassa 2.



Kuva 2: Anturointipiiirilevyn kaksi versiota. Ensimmäinen versio vasemmalla.

Anturointilevystä tehtiin toinen versio, jossa oli vain ATmega48-mikrokontrolleri ja vähän häiriöäntä tuottava regulaattori. Tämän levyn liitin on normaali 2,54mm:n 2x5 piikkirima. Levyn tosin laitettiin ylimääräinen rivi liittimen kiinnitysreikiä siten, että liitinriman voi laittaa kummalle puolen levyä tahansa pinnijärjestyksen pysyessä samana. Tätä toista anturointilevyn versiota käytettiin lopullisessa mittalaitteen prototyypissä. Levystä puuttui kokonaan mikrokontrollerin digitaalisen käyttöjännitteen häiriönpoistokondensaattori, mistä syystä piirilevystä piirrettiin kolmas versio, johon tämä puute oli korjattu. Kondensaattorin puuttuminen ei kuitenkaan aiheuttanut ongelmia käytössä, sillä jänniteregulaattori kondensaattoreineen on lähellä mikrokontrolleria ja niitä yhdistävät melko leveät kuparivedot.

2.4 Laitteen runko ja kotelointi

Runko ja kotelointi valmistettiin lopulliseen muotoonsa anturipiiirilevyjen ensimmäisen version jälkeen. Myös piirilevyjen kiinnitykseen tarkoitetut reiät porattiin tuota piirilevyversiota varten. Reikiä ei kuitenkaan tarvinnut porata uudelleen, sillä piirilevyjen kiinnitysreikien suhteelliset paikat pidettiin samoina versiosta toiseen.

Rungon materiaalivalinta tehtiin saatavilla olleiden materiaalien mukaan. Hiilikuitu olisi ehkä antanut kaikkein jäykimmän runkoratkaisun, mutta sitä ei ollut saatavissa riittävässä määrin, minkä lisäksi tarvittavia muotoja olisi ollut hankala tuottaa. Toisaalta JS Piirto Oy pystyi hankkimaan alumiinisia vatupasseja, jotka olivat alkuperäisen mittalaudan mittaisia ja muutenkin samankokoisia. Alumiini oli tietävästi myös vanhanmallisen mittalaudan materiaali. Lopputuloksen ei oletettu kuitenkaan olevan läheskään samankaltainen jäykkyys- ja muilta ominaisuuksiltaan kuin alkuperäisen mittalaudan, sillä mittalaitteen sisään tuleva elektroniikka kasvattaa sen painoa, minkä lisäksi jäykkyysominaisuudet muuttuvat muiden osien asennuksen myötä myös. Sen katsottiin kuitenkin olevan riittävän jäykkä mittalaudan runkona käytettäväksi.

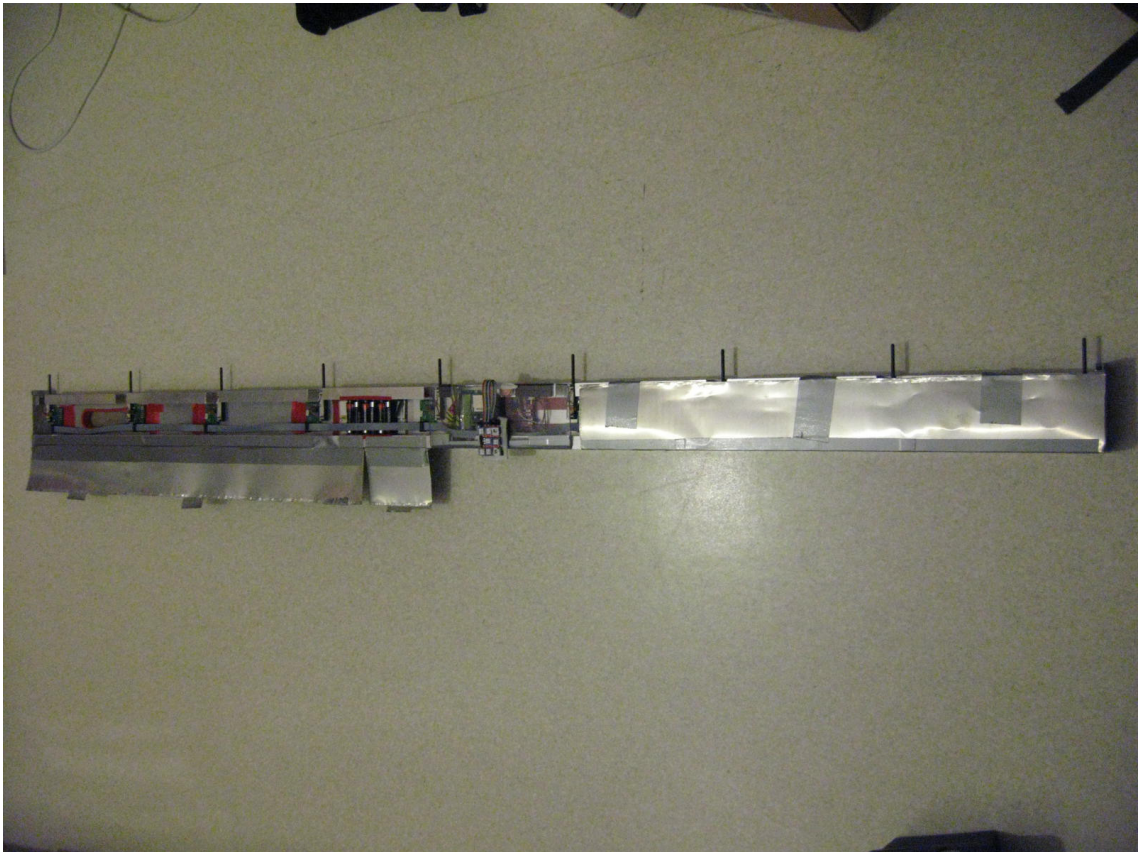
3 PROTOTYYPPILAITE

Tämä on kuvaus elektronisen mittalaudan prototyypin (myöhemmin ”mittalauta” tai ”mittalaite”) sähköisistä- ja rakenneratkaisista. Ohjelmistoa kuvataan kappaleessa 4.

Tässä kappaleessa on kuvia mittalaudasta sekä lopullisella että kehityksenaikaisella näppäimistöllä. Kehityksenaikainen piirilevy on esimerkiksi kuvassa 3. Kuviin ei ole erikseen merkitty kummasta versiosta on kyse. Prototyyppiin mekaaniset muutokset vanhempien kuvien ottamisen jälkeen rajoittuvat kuitenkin vain näppäimistöön ja sen piirilevyyn.

3.1 Yleisrakenne

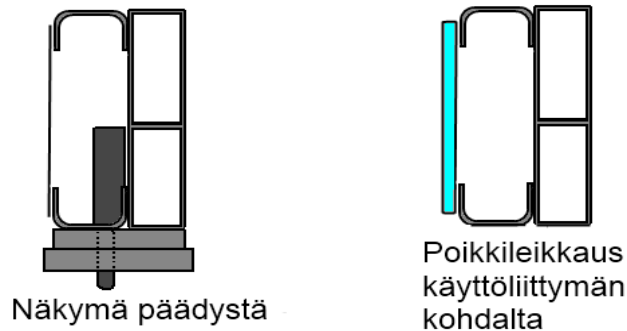
Rungon pohjana käytettiin alumiinista vatupassia, jonka pituudeksi valmistaja ilmoittaa 2m, ja jonka pituudeksi mitattiin 185cm. Vatupassi valittiin, koska se oli tarpeeksi



Kuva 3: Prototyyppilaite

jäykkä pysyäkseen mittausten kannalta riittävän suorana rasituksen alla. Se on myöskin riittävän suora, jotta kaikki anturit pysyvät samalla toiminta-alueella, kun niiden kiinnityspisteiden paikat asetetaan rungon alareunan mukaan. Toisin sanoen, runko ei aiheuta ylimääräisiä rajoituksia laitteen käytölle epäsuoruudellaan.

Vatupassiin kiinnitettiin sähköisiä komponentteja suojaamaan U-palkkia, mitat 20x30x20mm, molemmille pitkille sivuille, sekä näiden päälle 0,3mm alumiinilevyä. Laitteen näytön ja näppäimistön kohdalta alumiinilevy korvattiin akryylilevyllä, jotta näyttöä ja muita hallintalaitteita voidaan havainnoida. Näkymä päädyistä sekä poikkileikkaus ovat nähtävissä kaviosta 1.



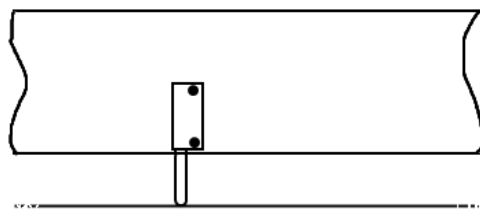
Kaavio 1: Rungon poikkileikkaus

3.2 Anturien sijoittelu

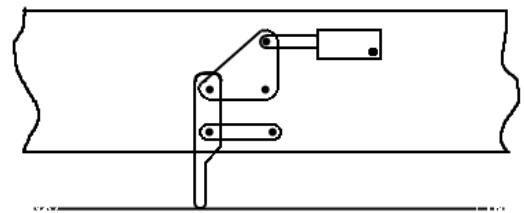
Anturit sijoitettiin mittalaitteen koko matkalle siten, että reunimmaisets anturit olivat 25mm rungon päädyistä ja muut tasavälein niiden väliin jäävällä matkalla. Tämä sijoittelu on RT-kortin mukainen mittaushohtien sijoittelu. Päätypisteiden tarkoitus ei käy ilmi RT-kortiston kortista. Ne kuitenkin otettiin mukaan sellaisenaan, sillä haluttiin pysyä RT-kortin mukaisessa mittaustavassa niin pitkälle kuin mahdollista. Tähän suuntaan kehityksen halusi tapahtuvan tutkimuksen tilannut yhtiö. Perusteluna tälle on rakennustekniikanyhteisön hyväksyntä mittaustavalle, jota ei siis haluta muuttaa enemmän kuin elektronisointia varten on pakko.

3.3 Mittalaitteen anturien fyysisen kytkentä

Koska mittaustavan katsottiin olevan kokeellinen niin anturien tarkkuuden kuin myös fyysisen kestävyuden osalta, päätettiin ensimmäinen prototyyppi valmistaa niin, että anturit ovat suorassa kontaktissa mitattavan pinnan kanssa, kaavion 2 tavalla. Tämä mahdollistaa sen, että mekaaninen linkitys mitattavalta pinnalta mittaushanturille ei vaikuta mittaustapahtuman tarkkuuteen. Toisaalta anturit ovat näin alttiita sivuttaisrasitukselle. Prototyypin odotetaan käytössä osoittavan kuinka paljon sivuttaisrasitus vaikuttaa antureihin. Pahimmassa tapauksessa anturit voivat fyysisesti katketa. Koemittausten aikana tätä ei kuitenkaan tapahtunut.



Suora yhteys mitattavaan pintaan



Epäsuora yhteys mitattavaan pintaan mekaanisen välityksen kautta

Kaavio 2: Suora ja välillinen anturin yhteys mitattavaan pintaan

Anturit kiinnitettiin runkoon ruuveilla. Niiden kiinnitys on vain yhteen levykerrokseen, mutta ne eivät ole siirtyneet koemittausten aikana. Tuotteistettaessa kiinnitys täytyy todennäköisesti suunnitella uudelleen joka tapauksessa, sillä sen täytyy pitää anturi paikallaan luotettavasti äärimmäisissäkin ulkolämpötiloissa, missä muovisen anturinkuoren ja nyt käytetyn metallisen ruuvien lämpölaajenemiskertoimien ero ylittää ruuvien kimmoisen venymän. Samassa yhteydessä voidaan uudelleenarvioida myös välitys.

3.4 Antuointipiirilevy

Anturin jännite luetaan Atmel Mega48V-mikrokontrollerin omalla AD-muuntimella, joka on 10-bittinen. Sama mikrokontrolleri hoitaa myös kommunikaatiota keskusyksikön kanssa.

Käytetyn mikrokontrollerin havaittiin keskeyttävän I2C-väylän toiminnan mikäli sen palvelurutiinissa yritetään vaivuttaa prosessori lepotilaan keskeytykset sallittuna. Syytä tähän toimintaan ei löydetty. Ongelma päätettiin kiertää suorittamalla AD-muunnoksia 100ms välein ja vaivuttamalla prosessori mittausten ajaksi ADC Noise Reduction -tilaan [9], joka sammuttaa kaikki prosessorin sisäiset kellosignaalit ja jättää toimintaan AD-muuntimen lisäksi vain ulkoiset herätelähteet, kuten I2C-väylän osoitetunnistuksen.

Koska prosessori herää vain joko ADC-muunnoksen valmistumiseen tai oman osoitteensa tunnistamiseen I2C-väylältä, ja koska keskeytysrutiinit viivyttävät muita keskeytyksiä suorituksensa ajan, mittaustulokseksi kopioidun AD-muunnoksen aikana ei tapahdu muuta prosessointia, joka voisi vaikuttaa mittaustulokseen. Näin ainoiksi häiriölähteiksi jäävät käyttöympäristön aiheuttamat häiriöt sekä I2C-väylän aiheuttamat häiriöt. I2C-väylän vaikutusta tuloksiin on vaikea arvioida tai mitata. Kontrollerin I2C-väyläohjaimen on kontrollerin datalehdellä luvattu olevan AD-muuntimen tarkkuuteen nähden häiriötä aiheuttamaton, mikäli muunnoksen näytettä ei oteta väylän pinnien lähettyvillä olevista pinneistä. Väyliä välinen sähkömagneettinen kytkentä taas pyritään minimoimaan pitämällä anturin ja AD-muuntimen välinen johdin mahdollisimman lyhyenä. Mikäli piirien välisen kommunikaatiioväylän liitintä

muutetaan, olisi mahdollista jopa juottaa mittaava piirilevy suoraan anturiin kiinni. Tätä ei kuitenkaan prototyyppilaitteessa tehty, sillä levyt ja anturit piti voida kehityksen aikana vaihtaa toisistaan riippumatta. Esimerkki prototyyppilaitteen lopullisesta anturointilevyn sovituksesta on nähtävissä kuvassa 4.

Anturointilevyn mikrokontrolleri kykenee sammuttamaan kaikki tarpeettomat moduulinsa virrankulutuksen ja sähköisten häiriöiden minimoimiseksi. Piirilevyn käyttöjännite on jaettu digitaalisiin ja analogisiin alueisiin, jotka on yhdistetty 10uH induktorilla. Impedanssin molemmin puolin välittömässä läheisyydessä on myös 100nF MLCC kondensaattorit. Tämä kytkentä on mikrokontrollerin datalehden suosittelema parhaan tarkkuuden saamiseksi. [9]

Mikrokontrollerin AD-muunnin ottaa käyttöjännitteensä analogiselta alueelta. Samaa jännitettä käytetään myös AD-muuntimen referenssinä, jotta matalataajuiset jännitteen muutokset, kuten esimerkiksi lämpötilan tai syöttöjännitteen vaikutus piirin jänniteregulaattoriin, eivät vaikuttaisi tulokseen.



Kuva 4: Anturointilevy kiinnitettynä runkoon

Regulaattorin ulostulon rippelijännite on 40uV - 400uV RMS, johon vaikuttaa valittu ohituskondensaattori. Regulaattorin sisäisen vahvistimen ohituskondensaattoriksi valittiin 100nF MLCC (koko 0805), jolla datalehden mukaan saadaan matalin

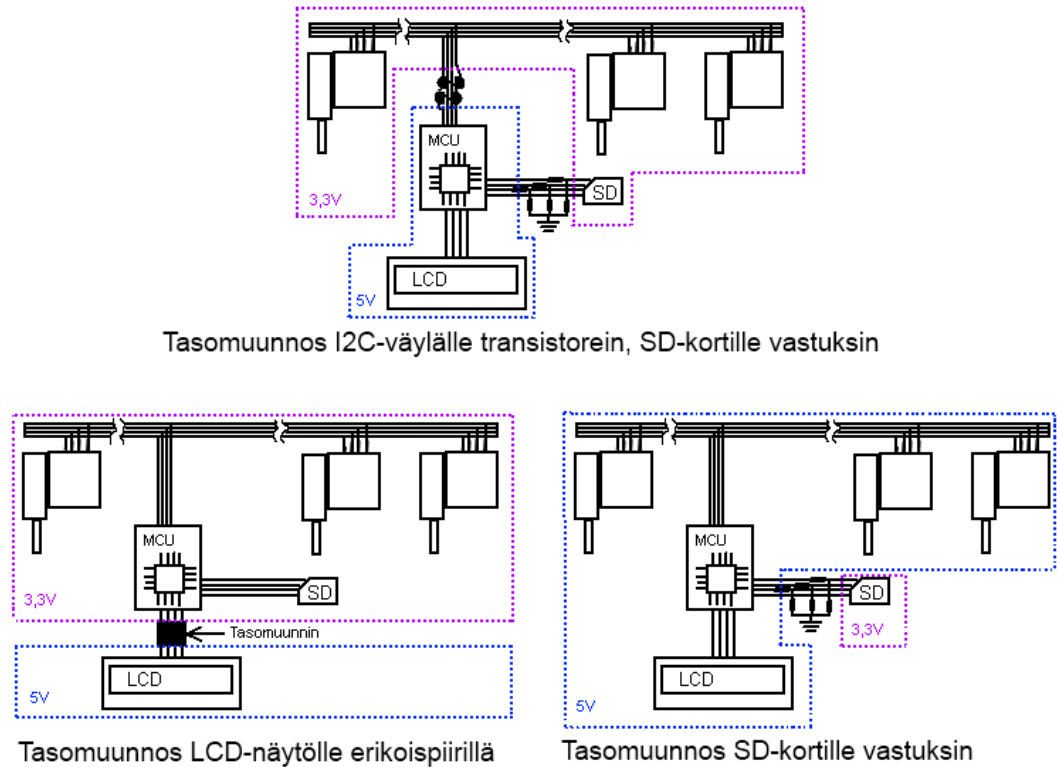
mahdollinen ulostulon rippeli. Ulostulon nousu hidastuu näin suurilla ohituskondensaattorin arvoilla myös, mutta anturilevyillä käytetyllä mikrokontrollerilla on sisäänrakennettu käynnistyksen hidastus [9], jolla mahdollinen ongelma ehkäistiin jo varmuuden vuoksi.

3.5 Laitteen käyttöjännite

Yleiseksi käyttöjännitteeksi valittiin 5V, koska kaikki käytetyt mikrokontrollerit ja LCD-näyttö toimivat sillä. Ainoa käytetty komponentti, joka ei toimi 5V jännitteellä, on SD-muistikortti, jonka käyttöjännitealue on 2,7V-3,6V. Muistikortin signaalit oli kuitenkin mahdollista sovittaa kontrollerilta kortille päin yksinkertaisella resistiivisellä jännitteenjaolla. AVR-sarjan mikrokontrollerit yleisesti hyväksyvät $V_{cc}/2$ -tasaisen signaalin loogisesti korkeaksi, joten muistikortin käyttämän SPI-väylän MISO-linja ei vaatinut lainkaan jännitetason muunnosta. Arduino-alustassa on sekä 5V että 3,3V regulaattori, joista jälkimmäisestä johdetaan muistikortille käyttöjännite.

Teoriassa mittausrakenteiden käyttöjännitteen muuttaminen 3,3V tasolle laskisi tehonkulutusta, mutta se aiheuttaisi myös signaalitason muunnostarpeen niiden käyttämälle I2C-väylälle. Tämä olisi melko helppo toteuttaa diskreetein transistorein, mutta sitä ei nyt tehty, sillä prototyyppi haluttiin pitää yksinkertaisena. Kommunikaatiioväylän kokonaispituus, 2,5 metriä, ja sen seurauksena valittu matala väylän taajuus sekä pituudesta johtuva väylän korkeahko kapasitanssi puoltavat tosin erillisten transistorien käyttöä siinä tapauksessa että tasomuunnos olisi tehtävä. Myös SD-kortille olisi edelleen tehtävä erillinen jännitetasojen sovitin.

Kolmas vaihtoehto olisi ollut käyttää kaikkia muita komponentteja 3,3V jännitteellä ja tehdä näytölle oma 5V regulointi sekä signaalitasojen muunnos. Munnettavia signaaleja olisi tosin tällöin ollut enemmän kuin niitä valitulla kokoonpanolla oli. Lisäksi tasojen sovitin olisi ollut haastavampaa, sillä käytetyn LCD-näytön loogisesti korkeaksi tulkitsema sisääntulevan signaalin taso on datalehden mukaan $0,8 \times V_{dd}$, eli 4V, mikä menisi tuolloin yli mikrokontrollerin käyttöjännitteen. Vaihtoehdot havainnollistettiin kaavioon 3.



Kaavio 3: Jännitealueiden vaihtoehdot

3.6 Mekaaninen suojaus

Muutokset ympäröivässä lämpötilassa voivat aiheuttaa kosteuden kondensoitumista laitteen sisälle, sillä antureiden tyyppin vuoksi kotelo ei voida tiivistää. Lisäksi se on riittävän suuri tilavuudeltaan saadakseen kondensaatiota aikaan jo sisältämästään ilmamassasta. Jotta kondensaatiovevettä ei pääse kerääntymään, päätettiin prototyyppilaitteen kansilevyjen reunoihin ja rungon pätyihin jättää aukot, joista ilma pääsee liikkumaan laitteen sisällä.

Mittapisteiden piirilevyt on myös suunniteltu pinnoitettavan jollakin paksuhkolla aineella, kuten esimerkiksi joustavaa täyteainetta sisältävällä epoksilla. Tämä suojaisi niitä myös pölyltä, metallilastuilta ja muilta laitteen sisään pääseviltä materiaaleilta.

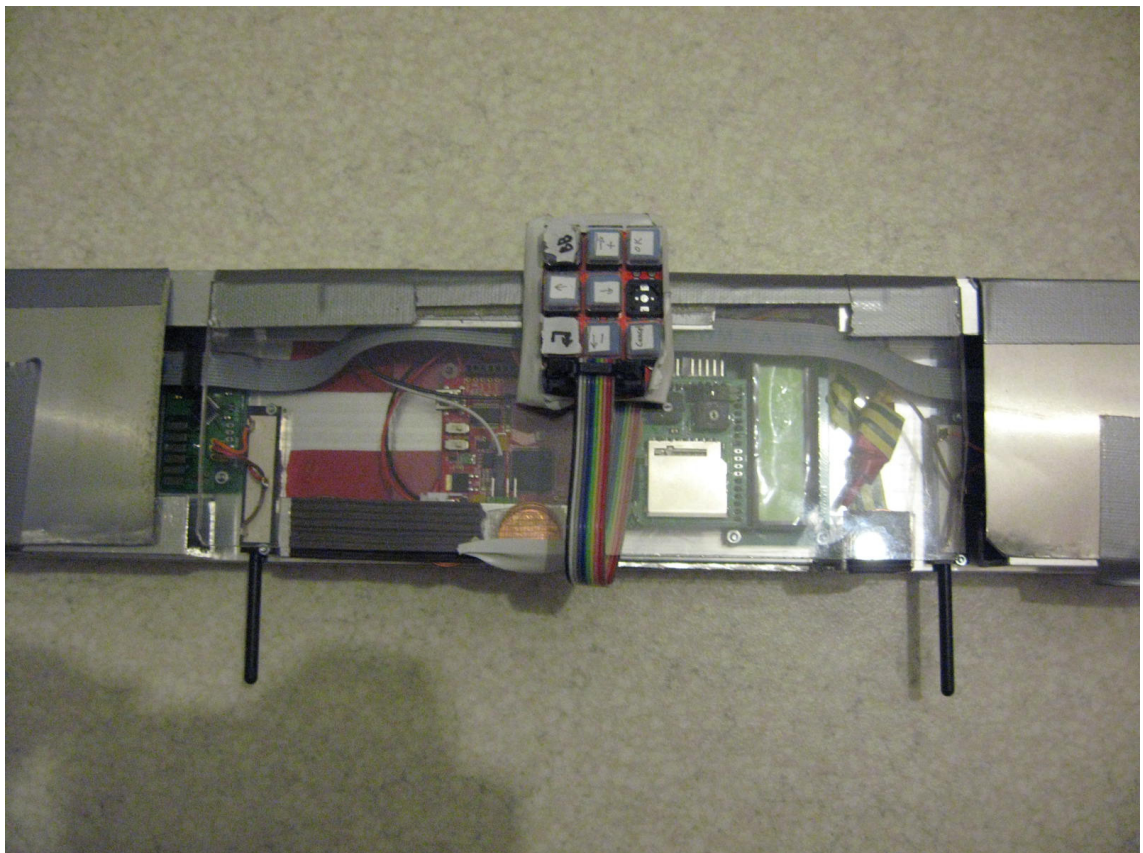
Käytettyjä antureita ei niiden rakenteen vuoksi voida päällystää täysin tai muuten kondenssi- tai pölysuojata. [10] Tämä on kuitenkin mahdollista kiertää. Rakennusteknisillä mittalaitteilla ei pääsääntöisesti saa muutenkaan tehdä mittauksia silloin, kun on vaara että niihin on kondensoitunut kosteutta. Tämän vuoksi ongelma voidaan minimoida asettamalla laitteen tarkkuuden ehdoksi kahden tunnin kuivatus ja kalibrointi kaikkien kylmästä lämpimään tilaan siirtojen jälkeen. Yhdistettynä avoimeen kotelorakenteeseen tämän pitäisi kuivattaa sensorit tarpeeksi, että niiden tulosten

toistettavuuteen voidaan luottaa. Muutoin mittalaitteelle voidaan asettaa käyttölämpötilarajoiksi sen osien vaatima käyttö ja säilytyslämpötilojen alue.

3.7 Käyttöliittymäkomponentit

Käyttöliittymän painikkeet muodostavat yksinkertaisen matriisinäppäimistön. Kehityksen ja ensimmäisen kenttätestin aikana käytössä oli kuitenkin kuvan 5 näppäimistö, jossa näppäimien lukemiseen tarkoitetuilla pinneillä on diodisuojaus, mikä estää matriisin ajamisen oikosulkuun riippumatta painikkeiden tai sisääntulon johtimien tilasta. Diodisuojauksen vuoksi näppäimistön ulostuloilla on myös alasetovastukset. Suojaus on kuitenkin tuotantolaitteessa turha. Matriisin oikosulkusuojaus on jo toteutettu myöskin mittalaitteen ohjelmistossa käyttäen hyväksi mikrokontrollerin ulostulojen korkeaimpedanssista toimintatilaa. Täysin ohjelmallisen matriisinhallinnan myötä myöskään ylös- tai alasetovastuksia ei tarvita, sillä mikrokontrollerin tuloilla on sisäinen ohjelmoitava ylössetovastus. Lopullinen käyttöliittymä on nähtävissä kuvassa 6.

Mittalaitteen näyttönä toimii 2-rivinen alfanumeerinen LCD-näyttö, jota ajetaan Hitachi HD 44700 -standardin mukaisella neljän datalinjan ja kolmen hallintalinjan konfiguraatiolla. [4] Näytön taustavalolle ei tehty erillistä hallintalaitetta, mutta

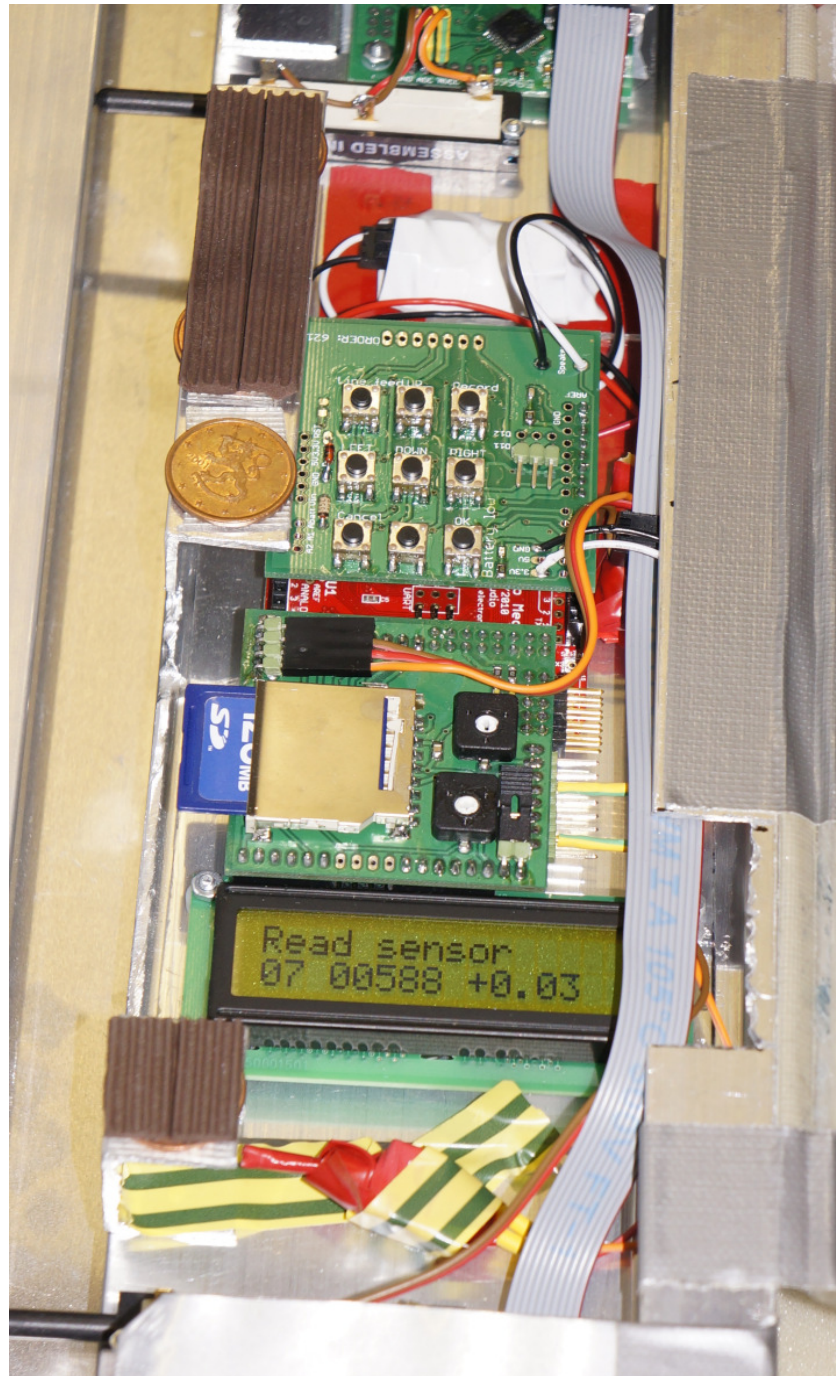


Kuva 5: Käyttöliittymä, ulkoinen näppäimistö

taustavalon ledeille asetettiin etuvastus ja 2,54mm:n liitinrimasta muodostettu hyppylanka, joka on tuotantomallissa helppo korvata taustavalon kytkimen johdoilla.

Prototyyppilaitteessa sekä näyttö että kehityksenaikainen näppäimistö on yhdistetty Arduino-alustaan saman piirilevyn kautta, jolla on myös muistikortin liitin. Näppäimistö kiinnittyy tähän piirilevyyn liitäntäjohdolla, jonka liittimeksi piirilevyllä on 2x5 osio 2,54mm:n piikkirimaa. Näyttö on liitetty piikkirimalla ja piikkiriman vastakappaleella, joista piikkirima on LCD-näytön juotosnastoihin juotettu ja vastakappale piirilevyyn. Lopullinen näppäimistö kiinnittyy Arduino-alustaa ja näyttöä yhdistävän piirilevyn ohi suoraan Arduinin päällä oleviin piikkirimaholkkeihin.

Kuvassa 6 näkyvä valkoinen paketti, johon kulkee näppäimistön piirilevystä kaksi johtoa, on piezosummeri. Se kiinnitettiin erikseen, koska sen jalkojen väli oli eri kuin piirilevyllä suunnitellun mallin, jota ei yllättäen ollut saatavissa.

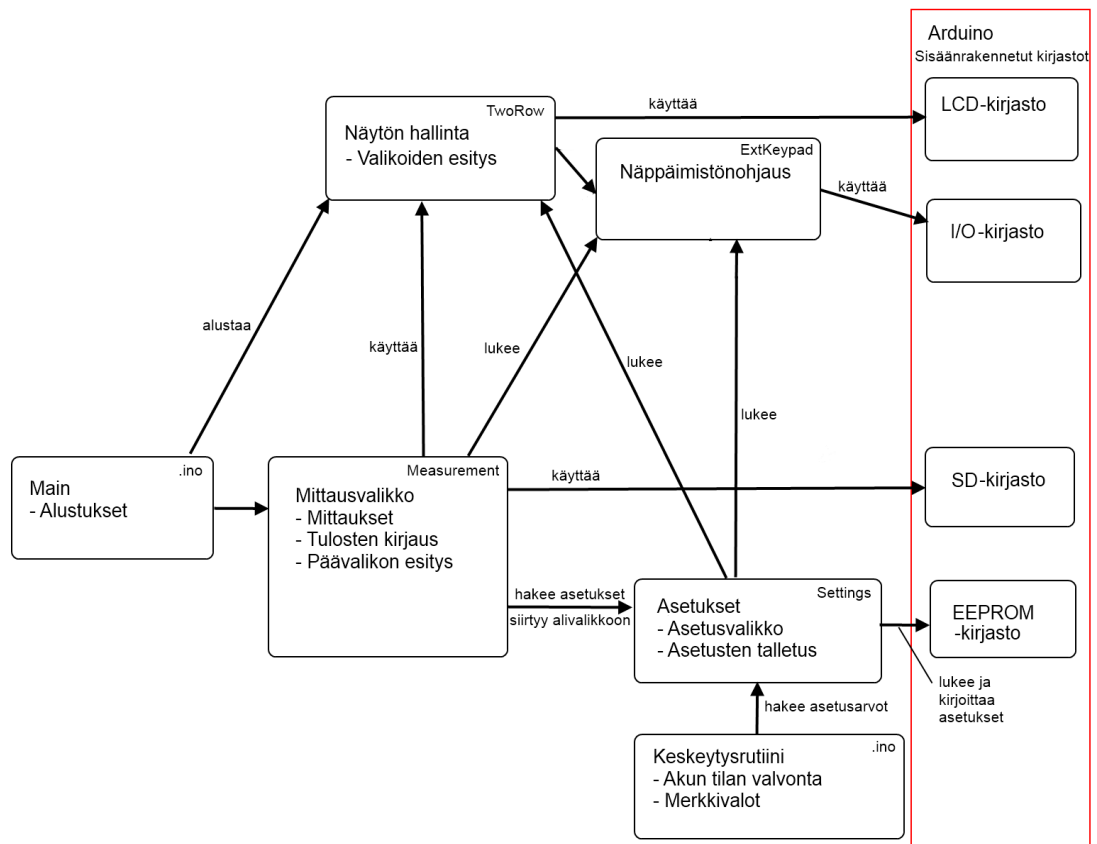


Kuva 6: Käyttöliittymä uudemmalla näppäimistöllä

4 PROTOTYYPILAITTEEN OHJELMISTO

Mittalaitteen ohjelmisto jakaantuu kahteen eri ohjelmaan. Toista ohjelmaa ajetaan laitteen keskusyksikössä, ja sen tehtäviin kuuluvat käyttöliittymän ylläpito ja tulosten keräys, muotoilu ja kirjaus muistikortille. Toinen, pienempi, ohjelma tehtiin mittausantureja lukeville mikrokontrollereille. Keskusyksikön ohjelma on käyttöliittymävetoinen; sen suoritus etenee vain käyttöliittymän kautta eikä muita prosesseja ole.

Keskusyksikön ohjelma kirjoitettiin Arduino-ympäristössä (versio 1.0.1). Ohjelmointiympäristö valittiin helpon saatavuuden ja nopean käyttöönotettavuuden vuoksi. Sitä mukaa kun ohjelmassa havaittiin yleistettäviä toimintoja, niitä koottiin kirjastoiksi. Kun mittalaite hyväksyttiin JS Piirto Oy:n puolesta valmistuneeksi ja toimintakuntoiseksi, oli tästä ohjelmasta eriytetty valikoiden tekoa yksinkertaistava luokka TwoRow, sekä asetuksia ja asetusvalikkoa kapsuloiva Settings. Näiden lisäksi omiksi luokikseen oli koottu myös näppäimistötoimintoja hoitavat funktiot, vaikka näin muodostunut luokka ei olekaan yleiskäyttöinen. Käyttöliittymän logiikan funktiot koottiin luokaksi Measurement, koska näin niiden yhteiset muuttujat oli helpompi alustaa. Ohjelman lohkokaavio on kaviossa 4.



Kaavio 4: Keskusyksikön ohjelman lohkokaavio

4.1 Käyttöliittymä

Käyttöliittymässä suorituksen yleinen kulku on seuraava: Ohjelma joko kirjoittaa näytölle kyselyn ja jää aktiivisesti pollaamaan näppäimistösyötteitä tai kutsuu valikkokirjaston funktiota, joka tekee saman ja palauttaa valitun valikkovaihtoehdon tai virhekoodin. Jokainen valikoiden taso pyrittiin tekemään omaksi funktiokseen, sillä valikoiden täytyi säilyttää tilatietoa. Suurin osa näistä funktioista on luokassa Measurements. Vain asetusten muuttamisen valikko on luokassa Settings. Virheilmoituksen ja tekstivalikon esittämiseen, sekä numeerisen arvon muuttamiseen on funktiot TwoRow-luokassa, mutta nämä vain palauttavat käyttäjän vasteen funktion kutsujalle. TwoRow-luokan tarjoamat näkymät ovat kaaviossa 5.

Valikkofunktio

<Valikon nimi>
 > <Valikon haara>

- Painike <OK> valitsee alemmalla rivillä näkyvän valikkohaaran
- Painike <Cancel> peruuttaa edelliseen valikkoon
- Painikkeet <Up> ja <Down> muuttavat valikkohaaran valintaa

Virheen ilmaisun funktio

Error:
 <Virhelähde>

- Painike <OK> kuittaa virheen

Asetusarvon muuttamisfunktio

<Muutettavan arvon nimi>
 <Arvo>

- Painikkeet <Right/+> ja <Left/-> nostavat ja laskevat muutettavaa arvoa
- Painike <OK> tallettaa muutoksen
- Painike <Cancel> peruuttaa tallentamatta

Vapaamuotoinen esitys

<Erikoisfunktion nimi>
 <Erikoisfunktion tila>

- Kutsuva ohjelma määrittää näppäintoiminnallisuuden
- Ensimmäinen rivi tekstiä
- Toinen rivi kutsujan tulostettava

Kaavio 5: Käyttöliittymän näkymät tyypillisine näppäintoimintoineen

Koska käyttöliittymä sisältää paljon tekstiä, jota ei haluttu alustaa muistiin ohjelman käynnistyessä, on teksti talletettu mikrokontrollerin ohjelmamuistiin kääntäjän avainsanalla PROGMEM jokaisen merkkijonon edessä. AVR-mikrokontrollerisarjalle on olemassa valmiit funktiot merkkijonojen hakemiselle ohjelmamuistista. Niiden käyttö valikkologiikan välissä olisi kuitenkin johtanut ohjelman huonoon luettavuuteen, joten niiden käyttö päätettiin rajata TwoRow-luokan sisälle. TwoRow sisältää näin kaikki funktiot, jotka käsittelevät ohjelmamuistissa olevia merkkijonoja tai näytölle kirjoittamista. Tästä muodostui sen lopullinen kapselointikerros. Koska joidenkin valikoiden piti kyetä esittämään dataa, joka ei ollut ohjelmamuistin merkkijonoja tai valikoita, lisättiin rajapintaan myös funktio ulkoisen muistissa olevan merkkijonon syöttämiseksi näytölle.

Hyvin yleinen toiminto tekstivalikoissa on monivalinta eri valikkohaarojen välillä. Tälle tehtiin TwoRow-luokkaan oma funktionsa, joka ottaa argumenteikseen osoittimen ohjelmamuistissa olevaan merkkijonoon, jota käytetään valikon otsikkona, osoittimen taulukkoon, joka sisältää valikon vaihtoehtoja esittävien merkkijonojen osoittimet, viitteen muuttujaan, johon käyttäjän valitsema vaihtoehto talletetaan, sekä valikon eri

vaihtoehtojen määrä. Funktio palauttaa joko nollan, jos käyttäjä valitsi jonkin vaihtoehtoista, tai miinus yksi, mikäli käyttäjä painoi peruutusnäppäintä.

Käyttöliittymän korkeimman tason valikossa on prototyyppilaitteessa vaihtoehdot mittausten suorittamiselle, kalibroinnin suorittamiselle, asetusten muuttamiselle sekä anturien lukemiselle. Mittausten suorittamisen valikossa on vaihtoehdot uuden mittaustiedoston avaamiselle, jo olemassa olevan tiedoston avaamiselle sekä tiedoston poistamiselle. Asetusten muuttamisen valikossa on vaihtoehdot mittausten tallennusmuodon vaihtamiselle sekä pariston tyhjentymisen varoitustason asetukselle. Näistä pariston tason varoitusta ei ehditty koskaan toteuttaa ja sen valitseminen valikosta ei aiheuta mitään toimenpiteitä tai vaikutuksia. Kaikista valikoista ja funktioista on mahdollista peruuttaa pois peruutusnäppäimellä. Asetuksen muuttamisesta tai kalibroinnista pois peruuttaminen tällä näppäimellä aiheuttaa sen, että muutoksia ei tallenneta. Mittaustapahtumasta pois peruuttaminen sen sijaan aiheuttaa vain mittaustiedoston sulkemisen. Mitään tietoa mittaustapahtumasta pois peruuttaminen ei hävitä. Mittauksen suorittamisen ja anturien testilukeman näytöt on kuvattu kaaviossa 6.

Mittaustapahtuman näkymä

Measurement
<Mittauksen tai rivin numero>

- Näppäin <Record> tallettaa anturien senhetkiset lukemat
- Näppäin <Line feed> merkitsee rivinvaihdon
- Mittauksen tai rivinvaihdon tapahtuessa toiselle riville tulee näkyviin mittauksen tai rivin järjestysnumero

Sensorien luvun näkymä

Read sensor
xx yyyyy +zz.zz

- xx = anturin numero
- yyyyy = anturin AD-muuntimen lukema
- zz.zz = millimetreiksi muutettu kalibraatioarvojen perusteella korjattu anturin lukema
- Näppäin <Up> näyttää edellisen anturin
- Näppäin <Down> näyttää seuraavan anturin
- Näppäin <OK> hakee uudelleen anturien lukemat

Kaavio 6: Mittaustapahtuman ja sensorien lukemisen näkymät

4.2 Asetusten ja kalibraatioarvojen talletus

Mikrokontrollerin EEPROM-muistiin talletetaan sekä yleiset käyttöasetukset että mittalaitteen anturien kalibraatioarvot. Näiden tietojen arvioitiin muuttuvan niin harvoin, että valmistajan EEPROM-muistille lupaama käyttöikä ei ylittyisi. Mikrokontrollerin datalehden mukaan AVR-sarjan EEPROM kestää 100 000 kirjoituskertaa. Vertailukohtana mekaanisten painikkeiden käyttöä voidaan usein olettaa enintään 100 000 – 1 000 000 painallusta.

Sekä Arduino-ympäristö että AVR-sarjan standardikirjastot tarjoavat funktioita EEPROM:n käsittelyyn. Koska käyttöliittymäohjelmisto nojautuu muutenkin Arduino-ympäristöön, päätettiin käyttää sen funktioita. Näin ohjelma on teoriassa helpommin siirrettävissä toiselle laitealustalle Arduino-ympäristön sisällä.

Arduinon EEPROM-kirjasto sisältää vain yksittäisten tavujen lukemisen ja kirjoittamisen, joten ohjelmakoodissa oli tarpeen iteroida talletettavan tai luettavan alueen yli. Tämä ei kuitenkaan muodostunut ongelmaksi, sillä erillisiä EEPROMIIN talletettavia ja sieltä luettavia alueita oli vain kaksi. Niille ei siis kannattanut kirjoittaa erillisiä talletus- ja lukufunktioita. Molemmat alueet kirjoitetaan kokonaisina. Muistin käyttöikää olisi voinut lisätä se, että niistä olisi kirjoitettu vain muuttuneet osat, mutta sen arvioitiin lisäävän ohjelman kompleksisuutta tarpeettomasti, hyötyjen jäädessä vähäisiksi. Muistin käyttöä rajaa joka tapauksessa useimmin muuttuva alue, joka mittalaitteen tapauksessa on kalibraatiodata.

EEPROM-muistin jako eri tallennusalueisiin tehtiin antamalla jokaiselle alueelle alustusfunktion argumenttina edeltävien moduulien alueiden yhteiskoko. Tätä varten jokaisen EEPROM:a käyttävän luokan yhteydessä oli tehtävä julkinen struct-rakenne, joka sisälsi kaikki talletettavat arvot, ja jonka koko voitiin täten selvittää sizeof-funktiolla. Tällä tavoin tehtynä varausalueiden kokoja ei tarvinnut päivittää kehityksen aikana niiden muuttuessa. On tosin huomattava, että asetukset eivät alueiden muuttuessa pysy tällöin alkuperäisillä paikoillaan EEPROM-muistissa, vaan kaikki asetukset on asetettava uudestaan, kun mittalaitteprototyyppi uudelleenohjelmoidaan. Tämä koskee myös kaikkia tulevia ohjelmistopäivityksiä. Mikäli mittalaite päättyy koskaan sarjatuotantoon, on sen EEPROM-muistialueiden alkukohdat kiinnitettävä. Tämä on kuitenkin helppo tehdä korvaamalla sizeof-funktiot vakioarvoilla käyttöliittymän käynnistysrutiinissa.

4.3 Muistikortille kirjoitus

Käyttöliittymäohjelma kirjoittaa muistikortille käyttäen Arduinon SD-kirjastoa. Numerot muutetaan C-merkkijonoiksi käyttäen sprintf-funktiota. Arduinon sprintf-toteutus ei kuitenkaan osaa käsitellä liukulukuja, sillä ne on käännösvaiheessa jätetty

pois tilan säästämiseksi. Koska Arduinon kirjastoja ei haluttu kääntää uudelleen, tehtiin funktio, joka laskee kalibraatiodatan perusteella anturien arvoista muunnoksen millimetreiksi. Funktio kuitenkin palauttaa viitteen kautta laskutoimituksen tuloksen liukuluvun lisäksi kymmeninä mikrometreinä, kokonaislukuna, joka on helppo pilkkoa kahden desimaalin millimetriesitykseksi sprintf-funktioon.

Koska tiedon esitystavaksi oli valittu CSV, kaikki kirjoitus voitiin tehdä alusta alkaen oikeassa järjestyksessä, eikä ollut tarpeen muuttaa tiedoston alkua edes jatkettaessa jo olemassa olevaa tiedostoa. Tulostiedostossa on kahdenlaisia rivejä: Mittaustapahtumat tuottavat rivin, jolla on kaikkien mittalaitteen anturien lukemat pilkulla erotettuna. Rivinvaihtopainikkeen painaminen tulostaa rivin, jolla on teksti "Line" sekä rivin numero. Mittaustiedosto voi näyttää esimerkiksi seuraavanlaiselta:

Line 1

000.19, -04.74, -06.79, -07.75, -08.15, -06.19, -04.77, -04.67, 000.33

000.45, -01.81, -05.00, -02.57, 000.78, 000.86, -02.32, -01.81, 000.30

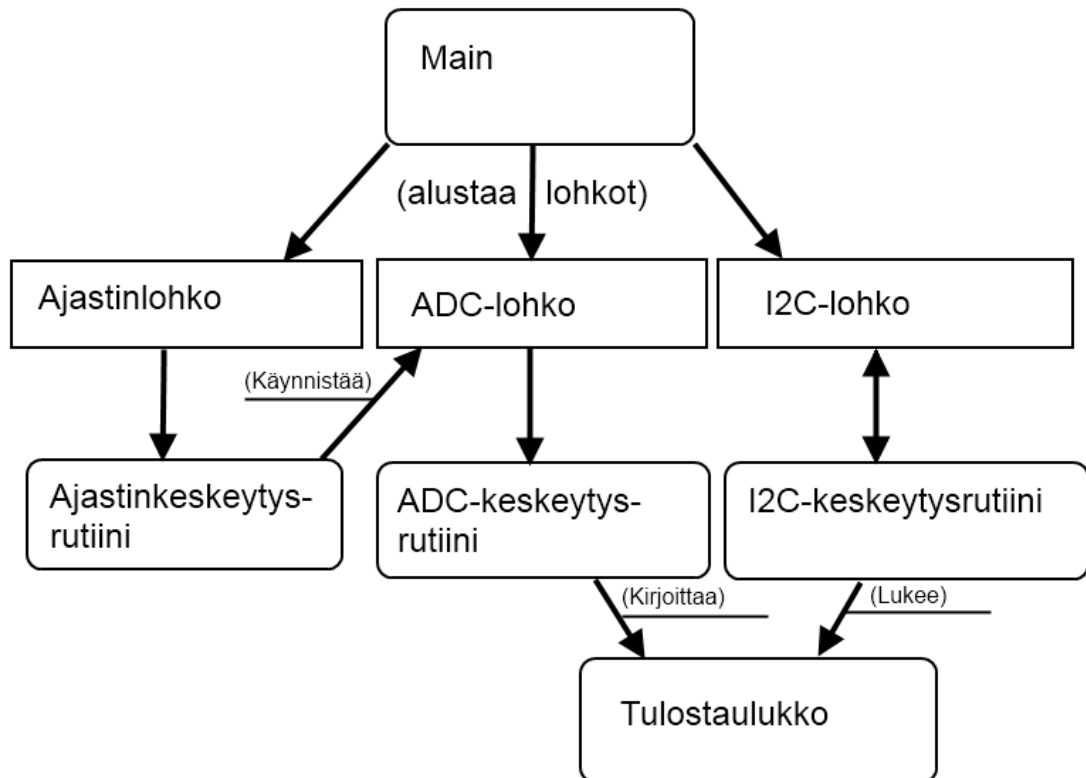
Line 2

000.15, -04.82, -04.65, -06.90, -08.31, -07.64, -06.25, -04.49, 000.37

000.65, -03.08, -05.64, -05.66, -04.90, -05.52, -07.09, -06.71, 000.41

4.4 Anturoinnin ohjelma

Antureita käsittelevien mikrokontrollerien ohjelma on liitteenä E. Se kirjoitettiin ja käännettiin Atmel AVR Studio -ympäristössä, käyttäen WinAVR-kääntäjää. Ohjelma koostuu vain neljästä eri funktiosta, mukaanlukien main-funktio; muut kolme ovat keskeytyskäsitteijöitä (ISR). Ohjelman lohkokaavio on kaavio 7.



Kaavio 7: Anturoidin ohjelman lohkokaaavio

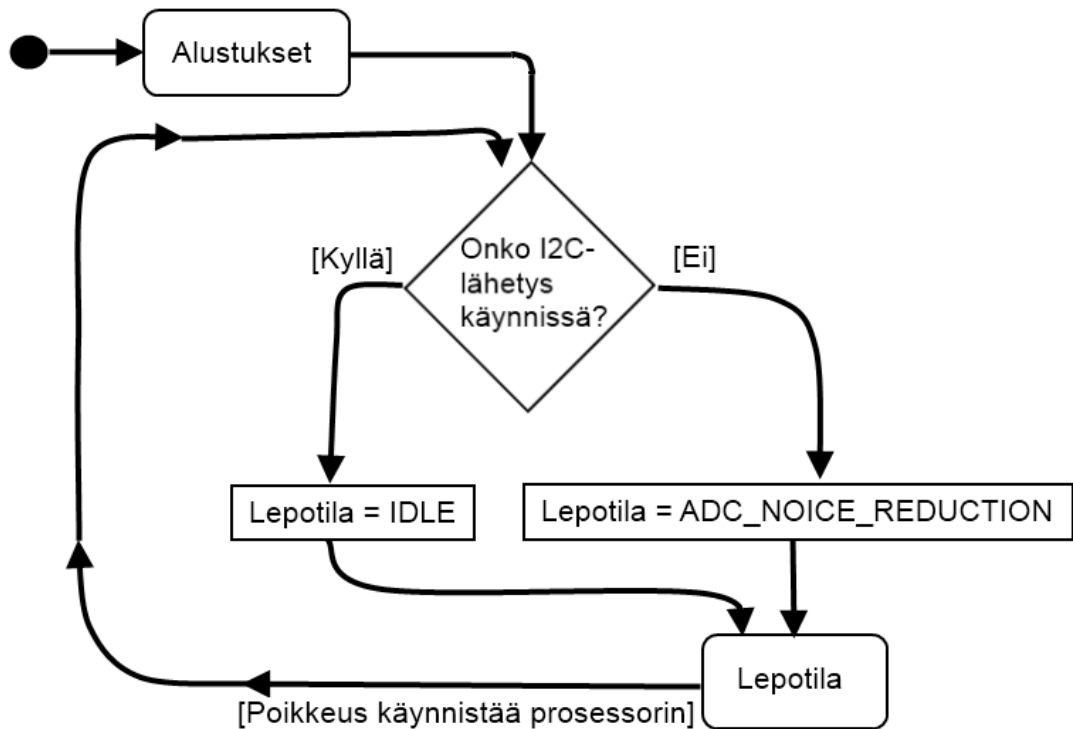
Suorituksen alkaessa main-funktio alustaa käytetyt oheislaitteet, minkä jälkeen sen suoritus jää ikuisen silmukkaan, jossa kontrolleri siirretään lepotilaan. Siirtymisen täytyy olla silmukan sisällä, sillä kaikki keskeytykset aiheuttavat lepotilan päättymisen. Lepotilan tyyppi riippuu I2C-väylän keskeytysrutiinin muuttamasta lippumuuttujasta, sillä syvempään lepotilaan siirtyminen keskeyttäisi meneillään olevan siirron väylällä. Ajastinlohko on asetettu laukeamaan säännöllisesti, ja se keskeytysrutiini käynnistää AD-muunnoksen. Kun muunnos on valmis, AD-muuntimen keskeytyskäsitteijä lukee AD-lohkon muunnostuloksen, laskee sille yksinkertaisen tarkistussumman ja kirjoittaa molemmat globaaliin taulukkoon. I2C-väylän keskeytysrutiini kopioi tämän tuloksen toiseen globaaliin taulukkoon ja lähettää tavu kerrallaan kommunikaatioväylällä.

ATmega48-mikrokontrollerissa I2C-keskeytysrutiiniin siirtyminen tapahtuu, kun kontrolleri tunnistaa oman osoitteensa tai on lähettänyt, vastaanottanut tai kohdannut virheen väylällä. Keskeytysvektoreita on vain yksi, ja keskeytyksen syy selvitetään lukemalla erikoisrekisteri. On huomattava, että suorituksen siirtyminen keskeytyksen palvelurutiiniin estää automaattisesti kaikki keskeytykset. Koska antureita lukevan ohjelman mikään palvelurutiini ei erikseen salli keskeytyksiä, voidaan olla varmoja, että globaalien muuttujien sisältö ei muutu osittain rutiinien aikana. I2C-väylän

palvelurutiini kuitenkin palaa jokaisen tavun lähettämisen jälkeen. Tästä syystä AD-muunnoksen tulos kopioidaan toiseen muuttujaan siirron alkaessa, mikä estää muunnostuloksen tavujen sekoittumisen seuraavan muunnoksen tuloksen kanssa.

Lepotila, jossa prosessori joutilaana aikana pidetään, vaihdetaan I2C-siirron alkaessa IDLE-tyyppiseen, sillä syvempään, AD-muunnoksen kannalta ideaaliseen ADC_NOICE_REDUCTION-lepotilaan siirtyminen keskeyttää kaikkien kommunikaatiolohkojen toiminnan. Syvempi lepotila otetaan käyttöön jälleen, kun siirto on valmis. Tämä on esitetty kaaviossa 8.

Lepotilan vaihto ei vaikuta niihin tuloksiin, jotka lähetetään keskusyksikölle, sillä tulosten siirtoon kuluva aika on suurempi kuin AD-muunnosten ottoväli. Näin useita uusia muunnoksia ehtii valmistua ajassa, joka keskusyksiköltä kuluu kaikkien anturien tulosten hakuun. Lisäksi, vaikka I2C-väylän nopeutta nostettaisiin, kuluu käyttäjältä mittalaitteen siirtämiseen mittausten välillä riittävästi aikaa, jotta väärän lepotilan aikana mahdollisesti tehdyn AD-muunnoksen tulos tulee ylikirjoitetuksi uudella.



Kaavio 8: Anturointilaudan pääohjelman tilakaavio

5 RESISTIIVINEN ETÄISYYSANTURI

Mittalaitteen vaatimukset ovat melko yksinkertaiset. Sen on on kyettävä vertaamaan kovan ja nimellisesti suoran pinnan todellisen tason muutoksia suorassa linjassa olevien, säännöllisesti sijoiteltujen antureiden avulla todelliseen suoruusjanaan. Tavoitteena oli mittaustapahtuman automatisointi, ja koska saatavilla ei ollut mekaniikkasuunnittelijaa, päätettiin jokaiselle mittapisteelle käyttää omaa anturia, joka olisi suoraan yhteydessä mitattavaan pintaan, ilman mekaanista välitystä.

Anturin lopullinen kiinnitys on esitetty kappaleessa 3.3, jossa myös kaavio 2. Tämä kappale käsittelee vain anturoinnissa käytettyä sensorelementtiä ja sen valintaperusteita.

5.1 Anturin tyypin valintaperusteet

Anturin tyypiksi valittiin resistiivinen lineaarinen jousikuormitettu anturi. Se on melko tunnoton heikoille magneettikentille, jopa siinä määrin että prototyypin kotelon kannen lukitusmekanismi on magneettinen. Resistiivinen anturi on myös melko helposti saatettavissa tunnottomaksi taustakohinalle ja sähköisille häiriöille normaaleilla piirilevyn häiriönpoistotekniikoilla. Koska sähkömagneettisen säteilyn siirtyminen on riippuva vetojen induktiivisuudesta ja pituudesta, riittää häiriöiden minimointiin yleensä johdinvetojen lyhyinä pitäminen. Lisäksi resistiivisen anturin lukemiseen vaaditaan vain AD-muunnin, joka on verrattain halpa komponentti.

Muita tärkeitä valintaperusteita olivat tunnottomuus pinnan materiaalille ja ainakin teoriassa myös käyttöympäristön pölyisyydelle. Puolivalmiissa rakennuksissa on yleisesti tavattavissa mm. sementtipölyä ja hienojakoisia metallilastuja esimerkiksi poraamisen jäljiltä. Optinen anturointi olisi altis pölyyntymiselle, mikä laskisi helposti mittalaitteen käyttöarvoa. Koska tutkittava betonipinta voi olla hyvin karkea, ultraäänianturoinnin tulokset voivat olla tarkkuudeltaan epävarmoja, mikä tekee niistä myös laadunvarmistustarkoitukseen hyödyttömiä. Lopuksi, minkäänlainen magneettinen anturointi laitteen kuoren ulkopuolella ei toimi, sillä rakennusten betoniosia vahvistetaan yleisesti teräksellä, joka ei ole välttämättä jakaantunut tasaisesti, mittalaitteen resoluution kannalta, tutkittavan betonipinnan sisään. Hall-anturilla toimivien potentiometrien käyttöä harkittiin, mutta ne olivat kalliimpia, mikä olisi tehnyt piensarjoista kannattamattomia tuottaa. Sitävastoin resistiivinen anturi on verrattain edullinen.

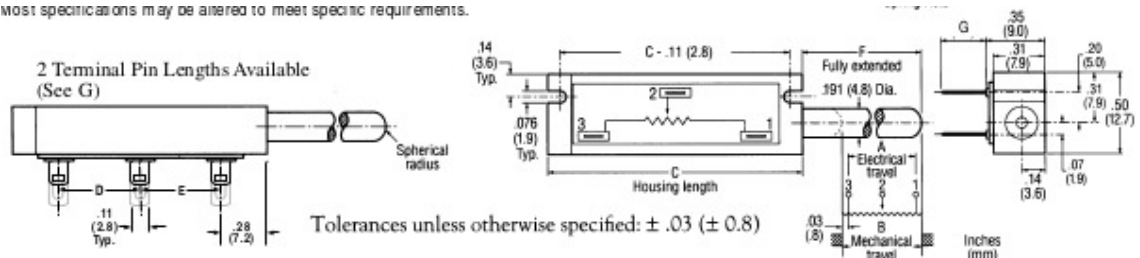
5.2 Anturin malli ja kuvaus

Valittu anturi oli BEI Technologies:n 9600 Series Miniature Linear Motion Position Sensor. (Jatkossa ”anturi”). Työssä käytetty malli oli 9615. [11] Se on esitetty kaaviossa

9. Anturimallin valinta perustui lähinnä saatavuuteen ja hintaan, sillä sen hinta, 40 euroa kappaleelta, oli noin kymmenesosan verran kilpailijoiden hinnoista. Se oli myös ainoa saatavilla ollut oikean kokoluokan resistiivinen anturi. Anturien hinnalla oli melko suuri merkitys, sillä kaupallista tuotantoa varten ne eivät voi olla hyvin kalliita kun niitä tulee laitteeseen 9 kappaletta.

Käytännössä kyseessä on lineaarinen potentiometri, jossa on sisäänrakennettu jousi. Prototyypissä käytetyssä mallissa on 39,6mm kokonaisliikerata, josta 38,1mm matkalta on mahdollista mitata muuttuva resistanssi. 35,6mm:n ”aktiivisella alueella” resistanssin muutos on potentiometrin asentoon nähden lineaarista. Valmistaja ilmoittaa anturin lineaarisuudeksi $\pm 2\%$ aktiivisen käyttöalueen yli ja parhaaksi käytännölliseksi lineaarisuudeksi $\pm 0,35\%$. Anturin kokonaisresistanssi on $5,1\text{K}\Omega \pm 25\%$. [11]

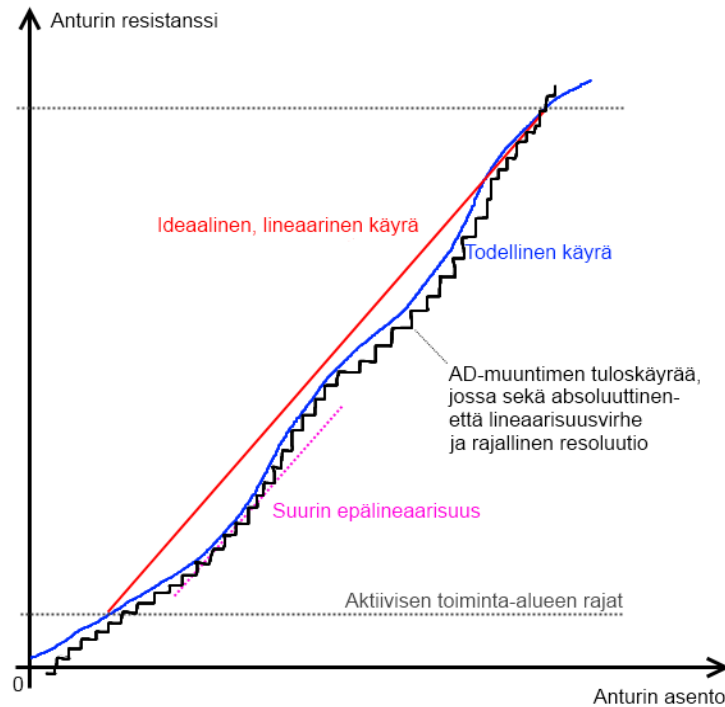
Most specifications may be altered to meet specific requirements.



Kaavio 9: Käytetty anturi (Lähde: [11])

5.3 Mittaustulosten tarkkuus

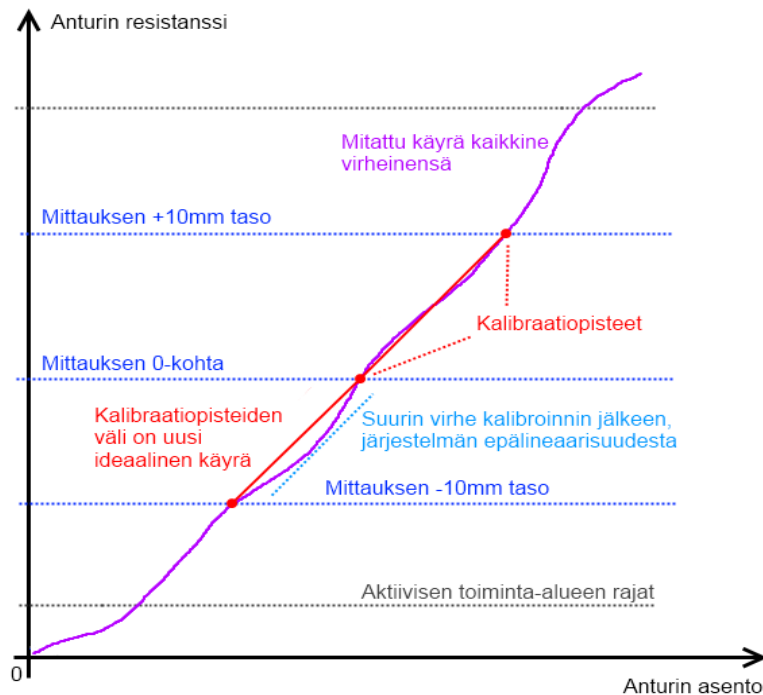
Potentiometrin resistanssi-asento -käyrä on teoriassa aina aidosti monotoninen, sillä normaalioloissa yksinkertaisessa vastuslangassa ei esiinny negatiivisen resistanssin kohtia tai täysin resistanssittomia osuuksia. [10] Resistanssi-matka -käyrän voidaan siis käytännössä olettaa olevan vähintään monotoninen. Anturien voidaan lisäksi olettaa olevan valmistajan ilmoittaman lineariteetin mukainen. Valmistajan ilmoittama $\pm 2\%$ epälineaarisuus koko anturin käyttöalueelta antaa tarkkuudeksi n. $\pm 0,8\text{mm}$. Näitä havainnollistaa kaavio 10. Tulosten tarkkuutta voidaan kuitenkin parantaa ottamalla näytteitä tunnetuista anturin asennoista ja linearisoimalla lukema näiden pisteiden välille, mitä havainnollistaa kaavio 11.



Kaavio 10: Anturoinnin virhelähteet

Prototyypilaitteeseen päätettiin laittaa kalibraatiotoiminto, jolla merkitään muistiin käytönaikainen nollapiste, sekä tästä pisteestä 10mm molempiin suuntiin. Mikäli kalibrointiin käytetty pinta on absoluuttisen suora, on mittaustarkkuus anturien osalta tämän jälkeen $\pm 0,2\text{mm}$. Tilastollisesti tarkkuus voi olla huomattavasti parempikin kalibraatiopisteiden läheisyydessä. Tällä ei kuitenkaan ole väliä, sillä rakennusteknisissä mittalaitteissa vain absoluuttisilla tarkkuusrajoilla on tulosten legitiimiyden kannalta merkitystä.

Anturin tuloksen mittaamiseen käytetylle mikrokontrollerin AD-muuntimelle on valmistajan datalehdellä annettu absoluuttiseksi tarkkuudeksi $\pm 2\text{LSB}$ ja integraaliseksi epälineaarisuudeksi 0.5LSB . Koska kaikki mittaustulokset linearisoidaan, ei mittaustuloksiin vaikuta muu kuin integraalinen epälineaarisuus. Mutta koska sillä ei ole anturin monotonisuusrajoitetta, voi virhe vaikuttaa kalibrointipisteissä ja mittaustuloksessa vastakkaisiin suuntiin. Siten se otettava huomioon kaksinkertaisena. Tällöin saadaan epätarkkuudeksi 10-bittisen AD-muuntimen osalta n. $\pm 0,04\text{mm}$. Suurin teoreettinen epätarkkuus laitteistolla on tulosten linearisoinnin jälkeen siis $\pm 0,24\text{mm}$.



Kaavio 11: Anturin tulosten tarkennus kalibraatiopistein

5.4 Mittausten toistettavuus

Potentiometrien toiminta perustuu mekaaniseen kosketukseen vastuslangan, -liuskan tai vastaavan ja liukuvan koskettimen välillä. Koska potentiometrin asennon muuttuessa liu'un on liikuttava vastusosan päällä, hankaus kuluttaa ajan myötä vastusosaa. Kun vastusosa kuluu, potentiometrin lukema samassa asennossa muuttuu. Kuluminen on teoriassa varmaa, mutta kulumisen nopeus ja aiheutuneet muutokset vaihtelevat potentiometrin rakenteen ja käytön mukaan. [10]

Koska mittalaitteen anturit ovat potentiometrejä, ovat ne myös alttiit tälle ilmiölle. Voidaan esittää oletus, että anturi antaa vähintäänkin suuntaa-antavia lukemia koko valmistajan ilmoittaman eliniän ajan, joka on noin 2 miljoonaa toimintasykliä. Koska kuitenkin mittalaitteelta vaaditaan valmistajan ilmoittamaa lineaarisuutta suurempaa tarkkuutta, on selvítettävä miten antureiden käyttö vaikuttaa niiden toimintaan.

Tätä vaikutusta testattiin suorittamalla mittalaitteella mittaustapahtuma siten, että anturin todellinen asento tiedettiin, minkä jälkeen anturia käsin liikuteltiin käyttöalueensa läpi 100 kertaa ja mittaustapahtuma toistettiin. Tätä tehtiin kymmenen kertaa. Tunnettuna asentona käytettiin kahta eri kalibrointiasentoa. Kahta eri asentoa käytettiin, koska ei ollut täyttä varmuutta siitä vaikuttaisiko kuluma enemmän käyttöalueen reunalla vai keskellä. Testiantureina käytettiin prototyyppilaitteen antureita 1 ja 2. Jokaisen mittaussyklin jälkeen lisättiin mittaustiedostoon rivinvaihtomerkintä, jotta mahdolliset virhepainallukset laitteen näppäimistöön eivät voisi heikentää tulosten

analysoitavuutta. Mittaukset tehtiin kolmesta kalibraatioasennosta vain kahdella, koska prototyypin kalibroitipalat olivat vielä kehityksen alaisia ja korkeimmassa kalibroitiasennossa hankalia käsitellä oikein.

Jotta tulokset olisivat mahdollisimman tarkkoja, talletettiin ne AD-muuntimen lukemina, muuntamatta millimetreiksi. Kokeen tulokset ovat liitteissä C ja D. Tulokset ovat kahdessa eri tiedostossa, koska koetapahtuma jouduttiin välillä keskeyttämään.

Tulokset anturien 1 ja 2 osalta kerättiin taulukkoon T1. Mikäli jostakin kulutuskohdasta otettiin useampi kuin yksi mittausta, otettiin tuloksista se, joka poikkeaa eniten edellisen mittauksen tuloksesta, sillä kokeessa etsittiin suurinta virhettä.

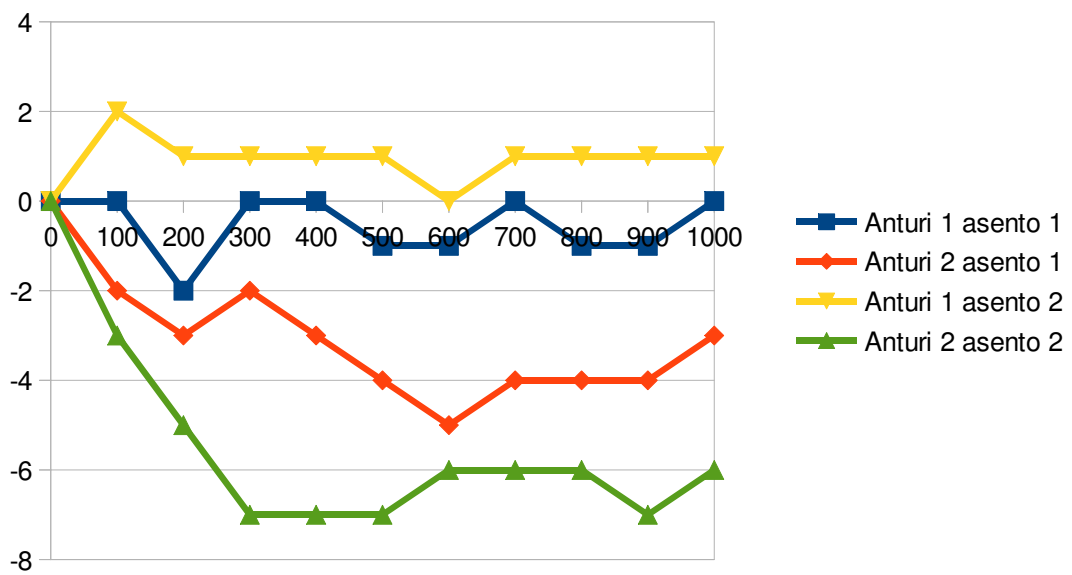
Käyttösyklejä	AD-muuntimen tulos			
	Anturi 1 asento 1	Anturi 2 asento 1	Anturi 1 asento 2	Anturi 2 asento 2
0	575	584	840	852
100	575	582	842	849
200	573	581	841	847
300	575	582	841	845
400	575	581	841	845
500	574	580	841	845
600	574	579	840	846
700	575	580	841	846
800	574	580	841	846
900	574	580	841	845
1000	575	581	841	846

Käyttösyklejä	Muutos alkutilanteeseen			
	Anturi 1 asento 1	Anturi 2 asento 1	Anturi 1 asento 2	Anturi 2 asento 2
0	0	0	0	0
100	0	-2	2	-3
200	-2	-3	1	-5
300	0	-2	1	-7
400	0	-3	1	-7
500	-1	-4	1	-7
600	-1	-5	0	-6
700	0	-4	1	-6
800	-1	-4	1	-6
900	-1	-4	1	-7
1000	0	-3	1	-6

Taulukko T1: Anturien 1 ja 2 toistettavuuskokeen tulokset ja muutos alkuarvosta

Taulukosta tehtiin käyrä K1. Käyrästä voidaan havaita, että anturien lukema muuttuu ensin kulutuksesta, mutta jo muutaman syklin jälkeen muutos pienenee yhden bitin kohinaksi.. Ilmiö on identtinen molemmille antureille. Kyseessä on todennäköisesti joko oksidikerroksen tai valmistuksessa jääneen pinnoitteen kulumisen pois. Mikäli kyseessä on oksidikerros, on se kulutettava pois aina ennen mittalaitteen käyttöä jokaisesta anturista, mikäli on vaara että se on päässyt uudelleen muodostumaan. On yhtäläisen mahdollista, että kyseessä oli anturien uutuudenkarheus, sillä niitä oli ennen tätä koetta käytetty vain joitakin kymmeniä toimintasyklejä kutakin.

Tämän perusteella todetaan, että sensorien tuloksen muutos rajautuu välille $\pm 1\text{LSB}$ silloin kun niitä on ensin käytetty niin, että epäpuhtaudet on kulutettu potentiometrin liuskoista. Täysi varmuus tästä saadaan kuitenkin vasta kun antureita testataan valmistajan lupaamaan käyttöikään asti sellaisissa määrissä, että tuloksille saadaan tilastollinen pätevyys. Tämä kuitenkin vain tarkentaa kuvaa anturien luotettavuudesta. Nyt saatujen tulosten perusteella uudet anturit antavat mittauskohdekohtaisen sisäänajon jälkeen vakaita tuloksia 500 käyttösyklin ajan, mikä JS Piirto Oy:n mukaan riittää käytännössä kaikille mitatuille lattiapinnoille.



Käyrä K1: Anturien tulosten muutos alkutilanteeseen kulutussykliä funktiona

6 PROTOTYYPILAITTEEN KENTTÄTESTIT JA TULOKSET

Kenttäkokeilla oli tarkoitus löytää prototyyppilaitteessa mahdollisesti jäljellä olevia vikoja ja virheitä. Niiden oli myös tarkoitus ohjata laitteen kehitystä ja viimeistelyä etenkin käyttöliittymän osalta. Mittalaitteen sujuva käyttö oli erityisen tärkeää, sillä pyrkimyksenä oli mittaustapahtuman keston ja virheiden minimointi.

6.1 Ensimmäinen kenttäkoe – Teollisuushalli Korpilahdella

Tämä ensimmäinen koemittaus tehtiin laitteella, joka oli juuri saatu ohjelmistollisesti ja mekaanisesti valmiiksi. Aikainen kokeiluhetki selittää osittain tuloksissa myöhemmin ilmenneitä systemaattisia virheitä; ohjelmiston toimivuutta ei oltu ehditty vielä erikseen testata. Mittaus tehtiin kiinteistön omistajan luvalla JS Piirto Oy:n suorittaman ilmastokosteusanturin asennuksen ja rakenteiden kosteuden mittauksen yhteydessä. Mitattava pinta oli teollisuushallin lattia, joka JS Piirto Oy:n edustajan mukaan oli hyvälaatuinen ja sopisi siksi erinomaisesti kontrollimittaukseksi.

Mittaustapahtuman tulosten toivottiin olevan samansuuntaisia mitatun pinnan nimellisen laadun kanssa. Varsinaisen mittaustapahtuman jälkeen suoritettiin kontrollimittaus eri tiedostoon. Tämä kontrollimittaus suoritettiin prototyypin kalibrointivälineillä, ja sen tarkoitus oli nähdä kuinka paljon mittalaudan sensorien lukema oli muuttunut kalibroinnin jälkeen.

Lopullista näppäimistöä ei saatu toimimaan ennen testin sovittua päivämäärää, joten testaukseen käytettiin kehityksenaikaista ulkoista näppäimistöä. Mittalaitteessa ei myöskään ollut asennettuna summeria eikä erillistä kytkintä näytteenotolle. Näytteenotto ja näyterivin vaihto oli ohjelmoitu yleisnäppäimistöön. Mittalaitteen summeri ei ollut kiinnitettynä, joten laite ei pitänyt minkäänlaisia ääniä. Mittalaite mittaushetkellä, esimerkkimittausta suoritettaessa, kuvassa 7.



Kuva 7: Mittauksen suoritustapa vastaa melko tarkalleen perinteistä tapaa

6.1.1 Mittaustulokset

Mittaustulokset ovat liitteessä A. Mittausten jälkeen suoritettun kontrollimittauksen tulokset ovat liitteessä B.

6.1.2 Mittaustulosten analyysi

Laite on tallettanut tulokset oikean muotoisina. Kontrollimittauksesta on kuitenkin nähtävissä, että kalibrointivälineillä saadut korkeudet ovat vain noin kolmasosan kalibrointipalojen todellisesta korkeudesta. Virhe on niin huomattava, että sen pääteltiin johtuvan ohjelmistovirheestä. Virhelähteeksi rajattiin lopulta lukualueen ylivuoto. Anturit mitataan 10-bittisellä AD-muuntimella, jonka tulos lähetetään keskusyksikölle vasemmalle tasattuna 16-bittiseen muuttujaan. Tätä muuttujaa käsiteltiin myöhemmin tavalla, joka salli lukualueen ylityksen.

6.1.3 Käyttötestin palaute

Mittalaudan todettiin tarvitsevan ulkonevat, mahdollisesti pidennettävät kahvat työergonomian edistämiseksi. Prototyyppilaitteessa oli kuitenkin jo olemassa kiinnitysmahdollisuus kyseisenlaisille apuvälineille, joten tämä palaute ei aiheuttanut jatkotoimenpiteitä.

Koekäyttäjä ilmoitti myös haluavansa näytön laitteen päälle, jotta se olisi havainnoitavissa helpommin käytön aikana. Tämä on kuitenkin ristiriidassa laitteen säätämisen helppouden kanssa, sillä helpon hallittavuuden vuoksi näppäimistön ja näytön olisi oltava nähtävissä samaan aikaan. Kuitenkin osa palautteesta on tarkoitus antaa käyttäjälle summerin kautta, ja koska se ei ollut laitteessa vielä tämän testin aikana, päätettiin näyttö pitää nykyisellä paikallaan kunnes päästään suorittamaan käyttötesti summerin kanssa.

6.1.4 Tulosten aiheuttamat toimenpiteet

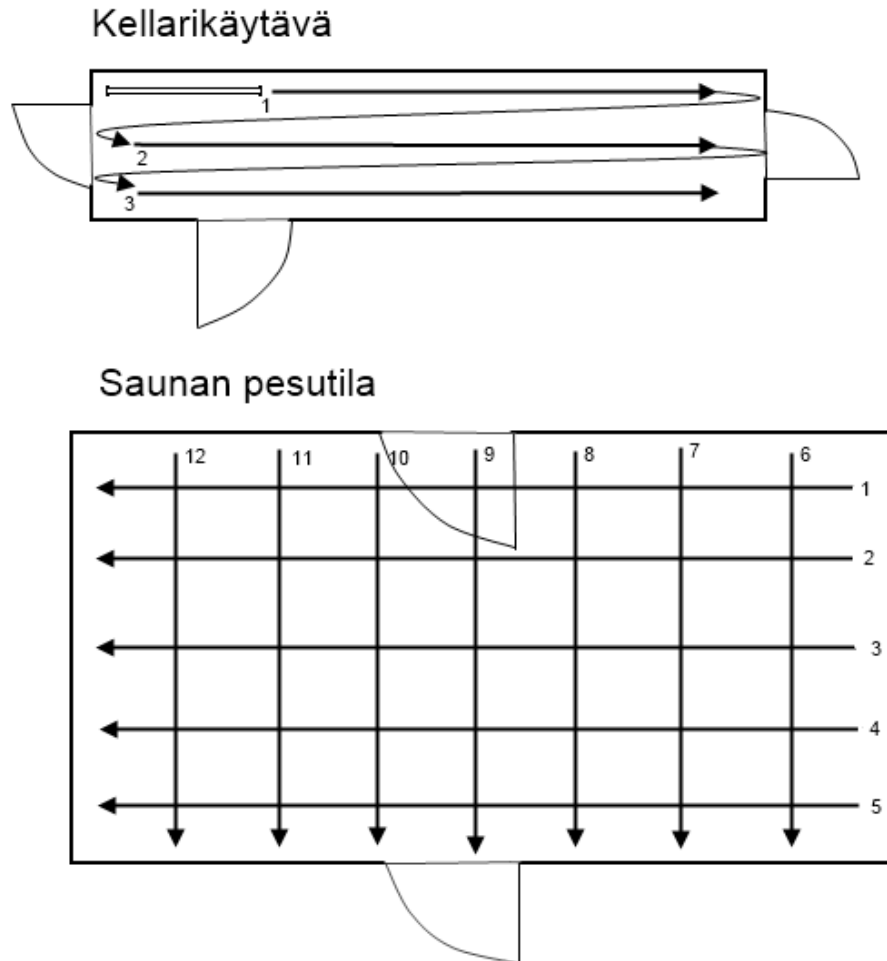
Havaittu ohjelmistovirhe korjattiin tasaamalla AD-muuntimen tulos pienempään bittiin päin, kun se tähän mennessä oli tasattu suurempaan. Lisäksi tulokset millimetreiksi muuttava funktio kirjoitettiin selkeämmäksi lukea, ja kaikki kertolaskut muutettiin tehtäväksi liukuluvuilla.

6.2 Toinen käyttötesti – Toimistokiinteistön kellaritiloja Jyväskylässä

Toinen käyttötesti tehtiin laitteella, jonka ohjelmasta oli korjattu edellisessä kokeessa esiintynyt laskentavirhe. Lisäksi sen näppäimistö oli vaihdettu uuteen, yksinkertaisempaan matriisinäppäimistöön, joka mahtui kotelon sisälle. Kalvonäppäimistöä luovuttiin kotelon kannen alle mahtuvan mekaanisen näppäimistön hyväksi, sillä sitä pidettiin varmatoimisempana ja luottamusta herättävämpänä. Lisäksi mittauksen suorittamiseen oli vedetty erillinen painonappi kotelon ulkopuolelle ja aiemmin suunniteltu summeri oli asennettu. Rivinvaihdon nappia ei oltu asennettu, vaan se oli edelleen näppäimistössä.

Mittauskohteena toimi Control Team Oy:n toimitilarakennus Jyväskylässä, osoitteessa Kypärätie 4. Näistä tiloista mitattiin kellarikäytävä ja saunan pesutilan lattia. Mittaussuunnat on merkitty kaavioon 12. Tilat oli jälleen valittu JS-Piirto Oy:n edustajan toimesta, jotta laitteesta saataisiin käyttöpalautta monenlaisista todellisista käyttöympäristöistä ja niiden ongelmista. Tämä ei ollut varsinaisesti linjassa kokeen nimellisen tarkoituksen kanssa, joka oli osoittaa laitteen toiminnan ja tulosten yhteneväisyys vanhan menetelmän vastaaviin. Toisaalta mittalaite on pakko kehittää

selkeästi vanhaa menetelmää houkuttelevammaksi mittauksen suorittajan näkökulmasta, sillä muuten se ei tule koskaan saavuttamaan kaupallista menestystä.



Kaavio 12: Mittaussuunnat toisen koemittauksen mittauskohteissa

Kontrollimittausta ei suoritettu varsinaisten kohteiden mittauksen jälkeen tällä kertaa, sillä mittautulosten toistettavuus oli tänä ajankohtana jo todettu erillisellä kokeella. Mittaustapahtumassa keskityttiinkin käytettävyyteen ja tulosten antamaan yleiskuvaan lattiapintojen muodosta. Mittalaite kuitenkin kalibroitiin ennen käyttöä sen kantokoteloon integroidulla kalibrointitasolla, kuten kuvassa 8.

6.2.1 Mittaustulokset

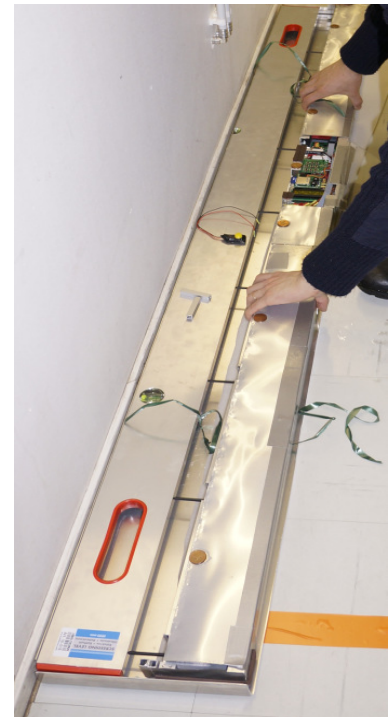
Mittauksen tulokset ovat tiedostoissa CONT1____.TXT ja CONT2____.TXT, joiden sisältö on liitteissä F ja G. Tuloksia luettaessa on huomioitava, että kellarikäytävää mitattaessa on unohdettu painaa rivinvaihtopainiketta ensimmäisen rivin jälkeen. Tuota mittausta luettaessa onkin oletettava rivinvaihdon tapahtuneen kuuden mittauskohdan jälkeen, viimeisen mittarivin pituuden perusteella.

6.2.2 Mittaustulosten analyysi

Kellarikäytävän mittaustuloksista on nähtävissä huomattavaa lattian aaltoilua sillä kohtaa käytävää, josta mittausrivit aloitettiin. Tämän aaltoilun paikka siirtyy hieman mittausrivistä toiseen, mutta sen olemassaololle muuten tasaisen käytävälattian toisessa päässä on olemassa järkeenkäypä selitys; rakennus on ollut useita vuosia käytössä ja tuossa päässä käytävää on kaksi ovea. Myös toisessa päässä käytävää esiintyy samanlaista kupruilua.

Saunan pesutiloista puolestaan näkyy selvästi kuinka lattiassa on kaareva kallistus veden johtamiseksi. Tuloksista ei kuitenkaan voida lukea kallistuksen suuntaa. Tämä olisi periaatteessa selvitettävissä kiihtyvyyssanturein, jolloin samasta mittauksesta saataisiin myös lattian kallistuskartta. Tuloksia on myös yhdessä tiedostossa paljon, ja niistä ei ole yhdellä silmäyksellä havainnoitavissa kovin paljon mitatun pinnan yleistilasta. Suurimman hyödyn saamiseksi tulokset pitäisi ohjelmallisesti visualisoida.

Molempien mittauspintojen tuloksissa voidaan havaita laitimmaisten anturien pysyvän positiivisissa lukemissa, vaikka muut anturit näyttäisivät tutkittavan pinnan olevan paikallisesti kuopalla. Tämän uskotaan johtuvan kosteuden kondensoitumisesta antureihin, sillä mittalaitetta ei voitu talvisäässä tapahtuneen ajomatkan jälkeen pitää lämpimässä tilassa riittävän kauan aikataulusyistä johtuen. Koska laitimmaiset anturit ovat lähellä mittalaudan päissä olevia korokkeita, näyttävät ne yleisessä tapauksessa nollatulosta tai hyvin pieniä arvoja. Tässä mittauskohteessa ainoa poikkeus tähän sääntöön havaittiin lähellä pesutilojen lattiakaivoa, missä lattian kallistus on suuri. Tästä voidaan päätellä, että kaikissa kirjatuissa lukemissa on olosuhteista johtuva systemaattinen virhe. Mittaustuloksista voidaan esittää arvauksia virheen suuruudesta,



Kuva 8: Kalibrointi toisen kenttäkokeen aikana

mutta varmasti se olisi mitattavissa vain asentamalla yksi pari antureita mittalaudan korokepalojen sisään.

6.2.3 Käytettävyyssanalyysi

Erillisen mittauspainikkeen ja summerin asentaminen nopeutti mittaustapahtumaa paljon verrattuna edelliseen kenttäkokeeseen. Kun edellisellä kerralla mittausta suorittava henkilö oli joutunut ajoittain varmistamaan näytöstä tulosten kirjaamisen, asennettu summeri teki tämän tarkistustoimenpiteen tarpeettomaksi.

Näppäimistön sijoittaminen mittalaitteen sisään kannen alle helpotti hieman laitteen käsittelyä, mutta toisaalta myös rinvaihtopainike oli nyt kannen alla. Laitteen suojakansi oli siis avattava aina kun mittausrivi vaihtui, minkä todettiin olevan epäideaalista. Uuden näppäimistön piirilevyllä on tosin liitosmahdollisuus ylimääräiselle painonapille, joten tämä puute voidaan korjata suhteellisen helposti.

Suurimpana käytettävyyden ongelmana oli tällä kertaa mittalaitteen suuri koko. Tämä ei ole varsinaisesti vika vaan ominaisuus, sillä mittalaitteprototyyppi on saman pituinen kuin perinteisen mittausmenetelmän lauta. JS-Piirto Oy:n edustaja pitikin sitä merkittävänä parannuskohteena, mikäli muita korvaavia mittausmenetelmiä joskus tutkitaan; ei nykyisen prototyypin vikana. Mittalaitteen pituus kuitenkin vaikuttaa merkittävässä määrin siihen millaisia tiloja ja missä suunnassa laitteella voidaan mitata. Esimerkiksi nyt mitattua kellarikäytävää ei olisi voitu mitata sivuttain. Myös mittalaitteen kuljetus mitattavaan tilaan voi olla ongelmallista silloin kun kuljetusreitti sisältää kapeita ja/tai matalia käytäviä sekä ovia.

6.2.4 Tulosten aiheuttamat toimenpiteet

Mittalaitteprototyypissä ei havaittu sellaisia vikoja, jotka välittömästi estäisivät sen käytön tuotantoympäristössä. Pariston varoitusvaloa ei oltu lisätty ohjelmaan vielä tätä koemittautusta tehtäessä, joten pariston tila oli ennen mittaustapahtumaa varmistettava yleismittarilla. Tämä on korjattava, mutta korjaus ei ole kiireellinen, joten se ei todennäköisesti ehdi tämän työn piiriin.

Mittalaitteelle on kuitenkin kirjoitettava käyttöohjeet, sillä kalibrointitapahtuma, yksinkertaisuudestaan huolimatta, on kriittinen laitteen tuottamien tulosten kannalta. Sama koskee myös ohjeita kondensaation estämisestä. Käyttöohjeet luonnosteltiin paperille ja annettiin JS-Piirto Oy:n arvioitavaksi.

7 TULOSTEN ANALYSOINTI

Tämä kappale käsittelee mittalaitteen, siinä muodossa kuin se on kuvattu kappaleessa 3, soveltuvuutta johdannossa kuvattuun mittaukseen. Prototyypilaitetta kuitenkin muutetaan yhä, kun mittalaitetta kehitetään edelleen.

7.1 JS-Piirto Oy:n huomioita

Prototyypilaitteen elektroniseen toimintaan oltiin tyytyväisiä. Myös käyttöliittymä on riittävän yksinkertainen. Mittalaitetta on kuitenkin yhä kehitettävä eteenpäin.

Suurimpana ongelmana nähdään JS-Piirto Oy:n edustajan mukaan hankala kalibrointi ja tarkaksi saattaminen antureita käsin kuluttamalla. Kalibroitirutiini nähdään hankalana, sillä sen on oltava tarkka, mutta toisaalta se on irtonaisilla korokepaloilla hankala yhden ihmisen suorittaa. Tätä pidetään yhtiön puolesta suurimpana ongelmana. Myös kalibroinnin tarkkuuden takaaminen tai edes virheen arviointi käytötavalle, jossa kertakalibroinnilla suoritetaan useita mittauksia pidemmän ajan kuluessa, olisi toivottavaa. Sen ongelmallisuus käytetylle anturityypille kuitenkin ymmärretään.

Toinen edustajan mielestä erityinen ongelmakohde on mittalaitteprototyypin kuljettaminen. Myös perinteinen mittalauta on vaikea kuljetettava pituutensa vuoksi. Nyt tehty prototyyppi kuitenkin nosti edelleen vaikeusastetta, sillä se on jossakin määrin herkkä kovalle tärinälle ja iskuille, joita esiintyy autokuljetuksessa työkalujen seassa. Lisäksi potentiaalisesti herkästi rikkoutuvat anturit ovat ongelma. Vaikka yhtäkään anturia ei saatu kenttäkokeissa rikki ja laitteelle tehty suojalaite, joka on nähtävissä kuvassa 8 mittalaitteen alla, suojaa antureita jonkin verran, kunnollisen kuljetusvälineen kehitys nähdään prioriteettina. Suojalaukun tai -kotelon tarve korostuu entisestään, kun ajatellaan tuotantoon siirtymistä.

7.2 Päätelmät

Anturien toistettavuuskokeen tulosten perusteella voidaan sisäänajetulle ja kalibroidulle prototyypimittalaitteelle antaa suurimmaksi epätarkkuudeksi $\pm 0,28\text{mm}$. Tämä on vanhaa mittaustapaa korvaamaan suunnitellulle laitteelle riittävä tarkkuus, ottaen huomioon esimerkiksi lattian tasaisuuden luokkien vaihteluväli. Suomen Betonilattiyhdistyksen luokituksen mukaan parhaan, A0-tasaisen, betonilattian sallittu poikkeama 2000mm:n matkalta on 4mm. Karkeimman, D-tason, lattian suurin poikkeama on 14mm.[12] Suurempaa tarkkuutta vaativiin mittauksiin on lisäksi RT-kortistossa määritelty erilainen mittaustapa. [1]

Saavutettu tarkkuus myös todistaa lineaarisen potentiometrin käyttökelpoisuuden tähän mittaustarkoitukseen. Tosin tämä soveltuvuusarviointi olisi toistettava, kun

prototyypilaitetta on käytetty aktiivisesti esimerkiksi vuoden verran. Tällöin saataisiin tietää anturien todellinen kulumisen käytön ja vuodenaikojen vaihtelun mukaan. Lisäksi potentiometrin voidaan katsoa olevan melko vaativa käyttöympäristönsä suhteen, sillä se vaatii kylmässä tai kosteassa tilassa säilytyksen jälkeen kuivatuksen ennen käyttöä. Tämä rajoittaa sen käyttömahdollisuuksia talvella, sillä mittauksia suorittavilla henkilöillä ei ole välttämättä aikaa odottaa talvella mittalaitteen kuivumista jokaisessa rakennuskohteessa joka on mitattava, kun mittalaite on tuotu ulkoa kylmästä.

Koemittausten perusteella prototyypilaitteen käyttöliittymä toimii luotettavasti ja laite nopeuttaa ainakin lattiapintojen mittausta. Vanhan menetelmän käyttö lähtökohtana kuitenkin aiheutti myös vanhan menetelmän heikkouksien, kuten pitkän laitteen vaikean käsiteltävyyden, periytymisen. Myös kalibroititapahtuma on hankala ja valitulla näppäimistöllä tiedostonimen kirjoittaminen syö tarpeettomasti aikaa, sillä kirjainten valinta on suoritettava yksitellen aakkosia läpi käyden.

Mittaustulokset on prototyypilaitteesta helppo siirtää muuhun ohjelmaan, kuten taulukkolaskentaohjelmiin. Kuitenkin suuri tietomäärä on vaikea visualisoitava. Tiedon erillisen purkamisen tarve ylipäättään on taka-askel vanhaan menetelmään nähden, sillä paperille kirjatessa oli mahdollista jo työmaalla mitatessa nähdä mitatun pinnan korjaustarve. Tutkittavaa pintaa ei silloin ollut välttämätöntä mitata kokonaan, vaan korjaus voitiin määrätä suoritettavaksi heti kun löytyy riittävä poikkeama. [13] Toisaalta, suurimman poikkeaman säilyttämiseen mittalaitteen muistissa mittaustapahtuman aikana ja asetusrajan ylittämisestä huomauttamiseen ei ole heti havaittavissa teknistä estettä.

7.3 Jatkokehitys

Vaikka lineaarisella potentiometrillä kyetäänkin suorittamaan mittauksia melko toistettavasti, on se kuitenkin mekaaniseen kosketukseen perustuvana jossakin määrin kuluva ja altis kosketuspintojen muutoksille. Lisäksi kalibroitirutiinin suoritus ennen jokaista mittauskohdetta kuluttaa tarpeettomasti aikaa. Näitä ongelmia voitaisiin yrittää korjata anturilla, joka mittaa asentonsa sisäisesti magneettisesti tai induktiivisesti. Tällöin kalibroitiväliä voitaisiin pidentää eikä käyttöolosuhteille ehkä tarvitsisi asettaa rajoituksia, kuten kuivatusaikoja. Lisäksi tarkemmalla anturilla varustettu mittalaite voisi korvata myös nykyään käytössä olevan, mittakelloon perustuvan tarkkuusmittauksen. [1]

Mittalaitteeseen voitaisiin myös lisätä muita mittauksia, kuten kallistus tai mitattavan pinnan lämpötila. Kallistusmittaus olisi hyödyllinen yhdessä tasaisuusmittauksen kanssa, kun halutaan varmistaa että lattia viettää koko alueeltaan kaivoon päin, eikä siinä ole kuoppia joihin vesi jäisi seisomaan. On myös huomattava, että betonilattian tasaisuus on määritelty poikkeamana ”vaakasuurasta tai nimelliskaltevuudesta” [12],

vaikka nykyään käytössä oleva mittaustapa nimenomaan ei huomioi kaltevuutta.[1] JS-Piirto Oy:n mukaan kaltevuus mitataan yleensä erikseen, mikäli tarpeen. Kiihtyvyyssanturin asentaminen mittalaitteeseen voisi siis nostaa sen käyttöarvoa huomattavasti. JS-Piirto Oy on ilmaissut halunsa tutkia tätä mahdollisuutta.

Mittalaitteen käyttöliittymää voitaisiin parantaa monella tapaa. Esimerkiksi tiedostonimen syöttäminen on nykyisellään hankalaa. Kun otetaan huomioon kosketusnäyttöjen viimeaikainen kehitys, voisi olla kannattavaa vaihtaa tähän mittalaitteeseen näyttö, jolla voitaisiin esittää graafisesti mittaustapahtuman edistyminen. Tällöin olisi kuitenkin tutkittava mitä tietoa on mahdollista hyödyntää jo mittauksen suorittamisen aikana ja miten. Potentiaalisesti mittauksen tuloksia ei tarvitsisi analysoida erillisellä tietokoneella, vaan mittalaitteen oma näyttö kykenisi esittämään tulokset jossakin helposti ymmärrettävässä muodossa ja talletettua tulostiedostoa tarvittaisiin vain todisteena myöhempää raportointia varten. Vaihtoehtoisesti voitaisiin tutkia mittaustulosten lähettämistä langattomasti älypuhelimelle ja sen näytön hyväksi käyttämistä kerätyn tiedon analysoinnissa.

LÄHTEET

[1] RT-Kortisto, kortti RT 14-10373, Rakennustieto OY, Rakennustietosäätiö, 1988, Suomi, 4 sivua

[2] Atmel Corporation, ATmega640/1280/1281/2560/2561 Preliminary, Datalehti, Rev. 2549L–AVR–08/07 , 2007, USA

[3] Arduino, Introduction, verkkosivu, Italia, <http://www.arduino.cc/en/Guide/Introduction>, viitattu 1.1.2013

[4] Shenzhen TOPWAY Technology Co., Ltd., LMB162ABC LCD Module User Manual, Rev. 0.2, Kiina (PDF, Saatavilla: <http://www.seeedstudio.com/depot/datasheet/LMB162ABC-Manual-Rev0.2.pdf> , 11.12.2012)

[5] Seeed Studio Wiki, Seeeduino Mega, [http://www.seeedstudio.com/wiki/index.php?title=Seeeduino Mega](http://www.seeedstudio.com/wiki/index.php?title=Seeeduino_Mega) (referoitu 8.12.2012, PDF-kopio saatavissa)

[6] SD Group & SD Card Association, SD Specifications Part 1: Physical Layer, versio 3.01, 18.5.2010 (PDF, Saatavilla: https://www.sdcard.org/downloads/pls/simplified_specs/Part_1_Physical_Layer_Simplified_Specification_Ver_3.01_Final_100518.pdf 1.1.2013)

[7] NXP Semiconductors, UM10204, I2C-bus specification and user manual, Rev. 5, 9.10.2012 (PDF, Saatavilla: http://www.nxp.com/documents/user_manual/UM10204.pdf 1.1.2013)

[8] Dangerous Prototypes, Where Labs, LLC. CC BY-SA , Bus Pirate: Wii Nunchuck quick guide, verkkosivu, <http://dangerousprototypes.com/2009/08/19/bus-pirate-wii-nunchuck-quick-guide/> , viitattu: 1.1.2013

[9] Atmel Corporation, Atmega48/V Preliminary, Datalehti, Rev. 2545T–AVR–05/11, 2011, USA

[10] Carl David Todd, Bourns, inc., The potentiometer handbook: users' guide to cost-effective applications, McGraw-Hill, U.S.A., 1975, 300 sivua

[11] BEI Technologies Inc. Duncan Electronics Division, Datasheet, 9600 Series Miniature Linear Motion Position Sensor, BE0026-9600-1001 , USA (PDF, Saatavilla: <http://www.beiduncan.com/pdf/9600datasheet.pdf> , 8.12.2012)

[12] Suomen Betoniyhdistys Ry (julk.nro. by 45) ja Suomen Betonilattiayhdistys Ry (julk.nro. BLY 7), Betonilattiat 2002, kirja, Suomen Rakennusmedia Oy, Suomi, 2002, 175 sivua

[13] Rakennustieto Oy, SisäRYL 2000 (Rakennustöiden Yleiset Laatuvaatimukset), kirja, Suomi, 1998, 416 sivua

LIITE A

Ensimmäisen koemittauksen (Teollisuushalli korpilahdella) tulokset:

-00.07, -00.38, -01.19, -01.30, -00.84, -00.09, 0.18, 0.29, 0.05

-00.27, -01.07, -01.88, -03.07, -04.06, -03.63, -02.39, -01.15, 0.02

0.03, 0.70, 1.71, 2.45, 3.01, 2.67, 1.57, 0.47, 0.02

-00.05, 0.14, 0.55, 0.67, 0.74, 0.46, 0.11, -00.19, 0.00

Line 1

-00.25, -00.42, -00.91, -01.00, -00.60, -00.09, 0.26, 0.76, 0.17

-00.37, -01.75, -01.88, -01.86, -02.27, -02.37, -02.01, -00.79, 0.00

-00.02, 0.53, 1.42, 1.93, 1.88, 1.21, 0.74, 0.39, 0.09

0.00, -00.15, 0.07, 0.61, 1.11, 1.32, 1.36, 0.70, -00.01

Line 2

-00.03, 0.10, 0.38, 0.76, 1.62, 1.72, 0.94, -00.23, -00.06

-00.25, -01.17, -02.29, -03.52, -04.44, -04.51, -03.58, -02.04, -00.14

-00.02, -00.30, -00.79, -00.61, -00.36, -00.13, 0.23, 0.09, 0.06

0.10, 1.20, 1.61, 2.09, 2.18, 1.39, 0.42, -00.10, 0.01

0.03, 0.44, 0.79, 1.01, 0.51, 0.05, -00.15, 0.09, 0.15

Line 3

0.01, 0.58, 1.21, 1.34, 1.23, 1.06, 0.63, 0.39, 0.06

-00.34, -03.28, -05.59, 5.99, 5.50, 5.72, 6.14, -03.73, -00.55

0.14, 2.06, 2.94, 2.23, 2.21, 1.96, 1.59, 1.00, 0.01

-00.01, 0.26, 0.63, 0.63, 0.35, -00.15, -00.27, 0.10, -00.02

-00.01, 0.02, 0.47, 0.98, 1.62, 1.67, 1.31, 0.37, -00.14

Line 4

-00.02, -00.19, -00.47, -00.05, 0.37, 0.81, 0.98, 0.42, -00.03

-00.19, -01.79, -02.92, -03.46, -03.77, -03.13, -02.49, -01.40, -00.13

-00.06, -00.10, 0.21, 0.79, 1.49, 1.22, 0.82, 0.37, -00.15

0.05, 0.98, 1.57, 1.54, 1.40, 1.14, 0.78, 0.29, 0.05

-00.09, -00.29, 0.30, 0.47, 0.39, 0.77, 0.67, 0.01, -00.25

LIITE B

Ensimmäisen koemittauksen (Teollisuushalli Korpilahdella) yhteydessä suoritettun kontrollimittauksen tulokset:

0.00, 0.00, 0.01, 0.02, 0.02, 0.02, 0.02, 0.02, 0.00

-03.57, -03.65, -03.53, -03.41, -03.49, -03.39, -03.48, -03.48, -03.53

3.43, 3.55, 3.64, 3.49, 3.61, 3.59, 3.68, 3.68, 3.64

LIITE C

Toistettavuuskokeen ensimmäisen tulostiedoston sisältö:

Line 1

00000, 00000, 00000, 00000, 00000, 00000, 00000, 00000, 00000
00575, 00584, 00587, 00616, 00622, 00604, 00598, 00584, 00577
00840, 00852, 00861, 00876, 00890, 00871, 00871, 00860, 00851

Line 2

00575, 00582, 00582, 00610, 00617, 00596, 00591, 00580, 00576
00842, 00849, 00853, 00866, 00879, 00857, 00857, 00848, 00843

Line 3

00573, 00581, 00581, 00609, 00617, 00597, 00594, 00584, 00582
00841, 00847, 00851, 00864, 00877, 00854, 00856, 00848, 00844

LIITE D

Toistettavuuskokeen toisen tulostiedoston sisältö:

00000, 00000, 00000, 00000, 00000, 00000, 00000, 00000, 00000

Line 1

00575, 00582, 00581, 00608, 00616, 00595, 00593, 00583, 00582

00841, 00845, 00846, 00859, 00872, 00850, 00852, 00847, 00845

Line 2

00575, 00581, 00579, 00606, 00613, 00593, 00589, 00580, 00578

00575, 00579, 00577, 00603, 00610, 00590, 00587, 00578, 00578

00841, 00845, 00847, 00859, 00872, 00849, 00851, 00846, 00843

Line 3

00574, 00580, 00579, 00606, 00613, 00592, 00588, 00578, 00576

00574, 00580, 00579, 00606, 00613, 00592, 00588, 00578, 00576

00841, 00845, 00847, 00860, 00872, 00851, 00853, 00847, 00846

00841, 00845, 00847, 00860, 00873, 00851, 00854, 00848, 00846

Line 4

00574, 00579, 00578, 00604, 00612, 00591, 00588, 00579, 00578

00840, 00845, 00848, 00861, 00874, 00853, 00855, 00848, 00847

00840, 00846, 00848, 00861, 00874, 00853, 00855, 00848, 00847

Line 5

00575, 00580, 00578, 00605, 00613, 00592, 00589, 00579, 00579

00841, 00846, 00848, 00861, 00875, 00853, 00855, 00848, 00846

00841, 00846, 00848, 00861, 00874, 00853, 00855, 00848, 00847

Line 6

00574, 00580, 00579, 00606, 00613, 00592, 00589, 00579, 00578

00575, 00580, 00579, 00606, 00613, 00592, 00589, 00579, 00578

00841, 00845, 00847, 00860, 00874, 00852, 00854, 00848, 00847

00841, 00845, 00847, 00860, 00874, 00852, 00855, 00848, 00847

00841, 00845, 00848, 00861, 00874, 00852, 00855, 00848, 00847

Line 7

00575, 00581, 00579, 00607, 00614, 00594, 00590, 00580, 00579

00575, 00581, 00579, 00606, 00614, 00594, 00590, 00580, 00579

00841, 00846, 00849, 00861, 00875, 00853, 00856, 00848, 00847

00841, 00846, 00849, 00862, 00875, 00853, 00856, 00849, 00847

LIITE E

Antureja lukevien mikrokontrollerien ohjelmalistaus.

```
// (c) Johannes Sulkakoski, 2012
// Free to use in projects, using this header if unchanged, or mentioning the source (me) in credits if not.
// No warranty of fitness for any purpose implied, conceived, etc..

// Purpose: This is a simple program that takes a sample from an ADC line whenever someone reads the microcontroller
// on the TWI (or I2C) line.

#include <avr/io.h>
#include <avr/interrupt.h>
#include <avr/sleep.h>
#include <util/twi.h>

// Any result has 3 parts: Upper byte, lower byte and checksum. Checksum is a XOR of the upper and lower bytes.
volatile uint8_t results[9];
volatile uint8_t resultsBeingSent[9];
volatile uint8_t resultSendCounter;

volatile uint8_t transmissionUnderway;

const uint8_t MYNAME = 0x09;

ISR(TIMER0_COMPA_vect)
{
    // The compare match interrupt flag is cleared automatically upon entering the interrupt handler.

    // Start an ADC conversion
    ADCSRA |= (1 << 6);
}

ISR(ADC_vect)
{
    uint8_t resultHigh;
    uint8_t resultLow;

    // The ADC result registers must be read in this order, because reading the ADCL will block writing to
    // both registers until ADCH is read also. This is by design, to avoid mixing results if
    // another conversion happens between reading them.
    // adcResultL = ADCL;
    // adcResultH = ADCH;
    resultLow = ADCL;
    resultHigh = ADCH;

    results[0] = resultHigh;
    results[1] = resultLow;
    // This byte is a poor man's checksum. It will catch most errors (since 1-bit-flipped is the most common one.)
    results[11] = resultHigh ^ resultLow;
}

ISR(TWI_vect)
{
    // Read the TWI status register and mask the clock prescaler bits away from the result
    uint8_t cause = TWSR & 0xF8;

    if((cause == 0xA8) || (cause == 0xB0))
```



```

{
    // --Someone wants to read our ADC result. Start a conversion and return results.--
    // Copy the last successful conversion results to our own send table and pass on
    resultsBeingSent[0] = results[0];
    resultsBeingSent[1] = results[1];
    resultsBeingSent[11] = results[11];
    resultSendCounter = 0;

    TWDR = results[0];

    // Mark the sleep logic to keep the sleep as idle, as any other sleep state will cause harm to the
transmission.
    transmissionUnderway = 1;
    set_sleep_mode(SLEEP_MODE_IDLE);
}
else if(cause == 0xB8)
{
    // --We have sent a byte (the MSB or the LSB of the ADC result). ACK has been received.--
    resultSendCounter++;
    TWDR = resultsBeingSent[resultSendCounter];
}
else if((cause == 0xC0) || (cause == 0xC8))
{
    // --We have received an expected NACK response. Nothing to do.--

    // After last byte the sleep mode can be the strictest possible again.
    transmissionUnderway = 0;
}
else if(cause == 0x00)
{
    TWCR |= 0b10010000;
}

// Clear interrupt flag (bit 7) by writing a logic one to it. Also re-enable TWI interrupts (bit 0), just in case.
TWCR |= 0b11000101;
}

int main(void)
{
    // --Initialize twi clock-- (remember clock prescaler restrictions)
    // (This is not really necessary, but we want to be prepared in case someone makes this program act as line master)
    // ((F_clk/F_target) - 16) / (2^4) = 14
    // 1MHz/(16 + 2*123*4) = 1kHz
    // 128k/(16 + 2* 14 *4) = 1kHz
    TWBR = 14;
    // TWPS1 = 0, TWPS0 = 1, -> Prescaler = 4
    TWSR = 0x01;

    // --Initialize TWI--
    // Write own slave address. The bit 0 is the general call accept bit. Leave it to zero, as
    // we don't use general calls for now.
    TWAR = (MYNAME << 1);
    // Enable in slave mode. Enable interrupts. Enable returning an ACK if called. Bits not in this order.
    TWCR = 0b01000101;

    // --Initialize ADC--
    // Technically, using the noise canceler to get ADC results as accurate as possible, the following should be
done:
    // a. Enable ADC in single conversion mode. Enable conversion complete -interrupt.
    // b. Enter ADC noise reduction mode (a special sleep/idle mode) or normal idle mode.
    //    Conversion should start automatically.
    // c. Handle ADC conversion complete interrupt.

    // Bits 7 and 6 here are reference selection. 01 for "AVCC with external capacitor at AREF pin".
    // Bit 5 is the left- or right-adjustment of results. For convenience, we take left-adjusted with a 1.
    // Bit 4 is reserved and reads as zero.
    // Bits 3-0 are ADC source selection. 0110 select ADC6.

```

```

ADMUX = 0b01100110;

// Bit 7 is ADC enable.
// Bit 6 starts conversion when written to 1. Write 0.
// Bit 5 enables auto trigger when 1. Write to 0.
// Bit 4 is the ADC interrupt flag. Write 1 to clear.
// Bit 3 is ADC interrupt enable. Write to 1 to enable the noise canceling also.
// Bits 2-0 are prescaler bits for ADC clock. 001 (or 000) sets a division factor of 2. ADC clock must be 50kHz -
200kHz.
ADCSRA = 0b10011001;

// --Initialize timer--
// Bits 7 & 6 are for affecting an output pin on OC0A compare match. 00 disables this functionality.
// Bits 5 & 4 are the same for OC0B
// Bits 3 & 2 are reserved and will read as zero
// Bits 1 & 0 are WGM1:0, Waveform generation mode. 10 is CTC; Clear Timer on Compare match.
TCCR0A = 0b00000010;

// Bits 7 & 6 are for forcing a compare match. 00 will do nothing.
// Bits 5 & 4 are reserved and will read as zero
// Bit 3 is WGM2 . It is 0 for CTC mode
// Bits 2-0 are clock source selection bits. 101 is clk/1024.
TCCR0B = 0b00000101;

// Timer CTC compare register is OCR0A
// We want measurements on 10 - 20 Hz, so 128kHz/1024/10Hz=
OCR0A = 13;

// Bit 2 is compare match B interrupt enable
// Bit 1 is compare match A interrupt enable
// Bit 0 is overflow interrupt enable
TIMSK0 = 0x02;

// Set sleep logic
transmissionUnderway = 0;

// Enable global interrupts
sei();

// Sleep until next interrupt
while(1)
{
    // Make sure ADC is enabled, ADC interrupt is enabled and
    // global interrupts are enabled and enter Idle mode.
    if(transmissionUnderway != 0)
    {
        set_sleep_mode(SLEEP_MODE_IDLE);
    }
    else
    {
        set_sleep_mode(SLEEP_MODE_ADC);
    }

    sleep_mode();
}
}

```

LIITE F

Tämä on mittalaiteprototyypin luoman tiedoston CONT1____.TXT sisältö. Sen esittämä mittaustapahtuma suoritettiin Kypärätie 4:n kellarikäytävässä:

Line 1

000.26, -01.11, 000.95, -02.53, -02.11, -01.25, -01.14, 000.90, 000.41
000.11, 000.50, -01.90, -03.65, -03.37, -03.92, -02.74, -01.92, 000.45
000.07, 000.46, 000.61, -02.89, -01.49, 000.07, 000.91, 000.52, 000.22
000.22, 000.54, 000.61, 000.25, 001.19, 001.25, 000.61, 000.86, 000.03
000.30, 000.07, 000.30, 000.16, 000.10, 000.62, 000.40, 000.36, 000.00
000.26, 000.07, 000.03, 000.12, 000.00, 000.11, 000.19, 000.79, 000.33
000.37, 000.53, 000.41, -01.32, -01.41, 000.25, 000.14, 000.07, 000.03
000.11, 000.46, -01.33, -02.28, -02.43, -02.98, -02.29, -02.03, 000.30
000.18, 000.46, 000.22, 000.16, 000.54, 000.03, 000.51, 000.88, 000.22
000.18, 000.11, 000.41, -01.00, 000.28, 000.03, -01.06, 000.94, 000.33
000.15, 000.03, 000.07, 000.92, -01.56, -01.45, -01.22, 000.30, 000.18
000.26, 000.46, 000.73, 000.62, 000.11, 000.50, 000.45, 000.70, 000.22

Line 2

000.30, 000.78, 000.92, 000.08, 000.93, 000.74, 000.73, 000.18, 000.11
000.03, -01.58, -02.06, -03.45, -02.78, -02.82, -02.44, 000.79, 000.07
000.30, -01.04, -01.10, 000.88, -01.80, 000.94, -01.52, -01.09, 000.11
000.22, 000.92, 000.64, 000.40, -01.37, 000.27, 000.72, 000.67, 000.03
000.18, 000.61, -01.10, 000.96, 000.07, 000.11, 000.69, 001.21, 000.29
000.07, 000.84, 000.80, 000.16, 000.28, 000.11, 000.34, 000.30, 000.18

LIITE G

Tämä on mittalaiteprototyypin luoman tiedoston CONT2____.TXT sisältö. Sen esittämä mittauksetapahtuma suoritettiin Kypärätie 4:n saunatilojen pesuhuoneessa:

Line 1

000.19, -04.74, -06.79, -07.75, -08.15, -06.19, -04.77, -04.67, 000.33
000.45, -01.81, -05.00, -02.57, 000.78, 000.86, -02.32, -01.81, 000.30

Line 2

000.15, -04.82, -04.65, -06.90, -08.31, -07.64, -06.25, -04.49, 000.37
000.65, -03.08, -05.64, -05.66, -04.90, -05.52, -07.09, -06.71, 000.41

Line 3

000.03, 000.03, -02.90, -05.62, -06.43, -07.56, -05.68, -02.00, 000.29
000.42, -03.39, -05.49, -09.71, -08.94, -08.43, -09.80, -06.90, 000.82

Line 4

000.07, -02.77, -04.35, -04.05, -03.41, -03.25, -01.25, 000.60, 000.30
000.15, -04.44, -09.92, -17.38, -16.70, -15.05, -13.28, -06.90, 000.26
000.45, -05.48, -06.18, -07.18, -04.82, -02.98, -02.51, -01.69, 000.37
000.34, -05.94, -11.14, -18.23, -23.76, -20.58, -14.04, -05.35, 000.71

Line 5

Line 6

000.11, 000.88, -02.02, -02.04, -01.01, 000.23, 000.43, 000.22, 000.22
000.19, -01.00, 000.76, -01.24, 000.82, -01.01, -03.39, -03.05, 000.75
000.00, -01.93, -04.92, -07.02, -07.56, -05.13, -04.12, -03.28, 000.33
000.38, -01.35, -04.88, -06.98, -06.31, -06.94, -05.11, -03.73, 000.56

000.15, 000.96, -02.13, -04.21, -06.62, -07.33, -09.61, -06.79, -01.16

000.26, 000.03, -02.63, -03.81, -05.49, -05.84, -05.49, -04.56, -01.05

000.22, -04.16, -06.75, -06.90, -08.31, -06.78, -05.26, -03.16, 000.41